Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра прикладної математики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Застосування ансамблевих методів для класифікації часових рядів.
(тема)

Виконав:
студент _2_ курсу, групи _ПМм-20-1
Федоров Д.П.
(прізвище, ініціали)
Спеціальність
113 Прикладна математика
(код і повна назва спеціальності)
Тип програми _освітньо-професійна
(освітньо-професійна або освітньо-наукова)
Освітня програма
Прикладна математика
(повна назва освітньої програми)
Керівник проф. Кіріченко Л.О.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ПМ _____ Тевяшев А.Д.
(підпис) (прізвище, ініціали)

2021 р.

Харківський національний університет радіоелектроніки

Факультет _інформаційно-аналітичних технологій та менеджменту
Кафедра _прикладної математики
Рівень вищої освіти _другий (магістерський)
Спеціальність _113 Прикладна математика
(код і повна назва)
Тип програми _освітньо-професійна
(освітньо-професійна або освітньо-наукова)
Освітня програма _Прикладна математика
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри ПМ _____
(підпис)
"____" _____ 2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _Федорову Демісу Петровичу
(прізвище, ім'я, по батькові)

1. Тема роботи _Застосування ансамблевих методів для класифікації часових рядів.

затверджена наказом по університету від 05 листопада 2021 р. № 1641 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 10 грудня 2021 р.

3. Вихідні дані до роботи _біологічні часові ряди індукції флуоресценції хлорофілу

4. Перелік питань, що потрібно опрацювати в роботі
1. Аналіз предметної області
2. Вибір і обґрунтування методу розв'язання
3. Програмна реалізація
4. Результати обчислювального експерименту
5. Аналіз можливих застосувань

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій

1. Актуальність теми роботи
2. Постановка задачі
3. Аналіз предметної області
4. Метод чисельного аналізу
5. Результати обчислювального експерименту

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Підбір та вивчення технічної літератури за темою роботи	8 – 14 листопада 2021 р.	виконано
2	Вибір та обґрунтування методу	15 – 21 листопада 2021 р.	виконано
3	Розробка алгоритму і програми	22 – 28 листопада 2021 р.	виконано
4	Проведення аналітичних досліджень та розрахунків	29 листопада – 5 грудня 2021 р.	виконано
5	Робота над текстом пояснювальної записки	6 – 9 грудня 2021 р.	виконано
6	Представлення роботи на рецензію в ЕК	10 грудня 2021 р.	виконано

Дата видачі завдання _8_ листопада 2021 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Кіріченко Л.О.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 50 с., 3 табл., 18 рис., 1 дод., 12 джерел.

ЧАСОВІ РЯДИ, КЛАСИФІКАЦІЯ, АДАПТИВНИЙ БУСТІНГ, БІОЛОГІЧНІ РЯДИ, МЕТРИКИ, ДЕРЕВА ПРИЙНЯТТЯ РІШЕНЬ, ІНДУКЦИЯ ФЛУОРЕСЦЕНЦІЇ ХЛОРОФІЛУ, АНСАМБЛЬ КЛАСИФІКАТОРІВ, ВИПАДКОВИЙ ЛІС.

Об'єкт дослідження – часові ряди індукції флуоресценції хлорофілу.

Метою роботи є побудова класифікатора часових рядів на прикладі біологічного ряду індукції флуоресценції хлорофілу.

Методи дослідження: беггінг та бустінг, ансамбль класифікаторів

У цій роботі проведено класифікацію біологічних часових рядів на прикладі показника індукції флуоресценції хлорофілу, експериментально отримані в ході дослідження набори даних, зібрані для аналізу, а потім відформатовані в робочий вигляд для зручного використання. Процес класифікації йшов за алгоритмами беггінгу та бустінгу, що дало нам досить точні результати.

ABSTRACT

Introductory note: 50 pages, 3 tables, 18 figures, 1 appendix, 12 sources.

TIME SERIES, CONSUMER, CLASSIFICATION, BIOLOGICAL SERIES, DECISION TREES, RANDOM FOREST, CHLOROPHYLL FLURESCENCE INDUCTION, ADAPTIVE BUSTING, METRICS, CLASSIFIER ENSEMBLE.

Object of research – time series of chlorophyll fluorescence induction.

Purpose of the work is to build a classifier of time series using the example of biological series, the chlorophyll fluorescence induction

Methods of research are bagging and boosting, ensemble of classifiers.

In this work, the classification of biological time series was performed on the example of the chlorophyll fluorescence induction, data sets were experimentally obtained in the course of research and collected for analysis and then were formatted into a working view for convenient use. The classification process went on bagging and boosting algorithms, which gave us quite accurate results.

ЗМІСТ

ВСТУП

Актуальність теми. Класифікація біологічних рядів буде потрібна завжди для вдосконалення аграрних процесів посіву та вирощування культур, якісного аналізу зібраних даних та подальших біологічних досліджень.

Часовий ряд – це розташовані у хронологічній послідовності числові значення статистичного показника, що характеризують зміну суспільних явищ у часі. Часові ряди мають значення для виявлення і вивчення закономірностей у розвитку явищ економічної, біологічних та медичних сферах суспільства. Звичайно, часові ряди класифікуються та можуть поділятися на декілька видів. На папері ряд можна розглядати, як купу точок даних, які послідовно розташовані на площині та рівновіддалені один від іншої. Також ряд можна описати як вид парної стохастичної залежності, котра має головний аргумент – час.

Кожний процес який повторюється повторно через фіксований період часу можна представити часовим рядом, зібравши комплекс спостережень. Наприклад, найбільш відомий економічний ряд – це курс валюти американського долара, який постійно зміню своє значення, зростає та падає із зміною часу, або в медичній сфері маємо кардіограму (ЕКГ), яка дозволяє діагностувати різноманітні види патології серця, визначає ритму серця та частоту серцевих скорочень.

Всі часові ряди поділяють на два класи: миттєві та інтервальні. В миттєвих рядах розглядаються показники в певні дати спостережень, а в інтервальному часовому ряді за певний інтервал або період часу.

Класифікація часових рядів дозволяє вилучити корисну неявну інформацію з набору даних для подальших раціональних дій або аналізу.

Мета і завдання кваліфікаційної роботи. Метою кваліфікаційної роботи є побудова класифікатору для виявлення неявних ознак з набору даних, розпізнання типу полива рослин по показнику ІФХ.

Отримана модель класифікатору повинна давати достатні результати точності класифікації часових рядів. У результаті на виході моделі ми отримаємо

дані, класифіковані за ознакою належності кожного ряду до конкретного типу.

Тож для аналізу зібраних спостережень потрібно побудувати якісний класифікатор, який дасть змогу отримати корисну інформацію з великого набору спостережень та вдосконалити процес землеробства, а саме допоможе у подальшому обприскуванні рослин аграріями та вилучення корисної інформації для подальшого аналізу росту рослин.

Отриманий класифікатор можна буде використовувати та вдосконалити у реальному житті після тестування на великих вибірках з різними факторами середи.

Для досягнення поставленої мети необхідно виконати наступні завдання:

– провести огляд і аналіз сучасного стану задачі «Класифікація часових рядів»;

– знайти предмет актуальний предмет дослідження;

– проаналізувати дата сети;

– провести аналіз задачі та розбити її на підзадачі;

– дослідити сучасні алгоритми класифікації;

– знайти програмну мову оптимальну для поставленої задачі;

– реалізувати алгоритм програмно та проаналізувати результати ;

– підвести висновки.

Об'єктом дослідження є часові ряди на прикладі біологічних рядів.

Предметом дослідження є виявлення класів поливу рослин за графіком індукції флуросценції хлорофілу.

Методи дослідження. У кваліфікаційній роботі використовуються ансамблеві алгоритми беггінгу та бустінгу, а саме RandomForest та AdaBoost.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1. Дослідження індукції флуоресценції хлорофілу

В сучасних умовах антропогенного впливу на оточуюче середовище, велике значення має екологічний моніторинг. Багатьма дослідниками показано, що вимірювання флуоресценції хлорофілу з найбільш швидким та інформативним методом, який використовується в екологічному моніторингу. Одним із методів отримання інформації про стан рослин є метод індукції флуоресценції хлорофілу(ІФХ). Він полягає у наступному: лист рослини опромінюють синім світлом, молекули хлорофілу збуджуються і починають випромінювати світло в червоній області спектра. Інтенсивність випромінювання залежить від стану рослини. Залежність інтенсивності випромінювання від часу називається кривою Каутського (кривою ІФХ), дана крива характеризує процес фотосинтезу.

Основна ідея цього методу полягає у тому, що хлорофіл, який знаходиться у фотосинтетичних мембранах, є природним датчиком стану клітин рослин. При порушенні стану фотосинтетичних мембран під впливом зовнішнього фактору починаються зміни оптичних властивостей хлорофілу, які є джерелом інформації для експрес-діагностики стану клітин рослин . Хімічні фактори та кліматичні умови впливають на параметри кінетики та спектральні особливості флуоресценції, а також на її стаціонарний рівень. Тому аналіз параметрів флуросценціі хлорофілу – це потужний інструмент дослідження впливу різноманітних екологічних факторів на рослини. З метою розробки методики оцінки стану рослин в реальному часі в Інституті кібернетики імені В.М. Глушкова НАН України в лабораторних умовах було проведено три експерименти, описані далі. Особливу увагу авторами приділено параметрам флуоресценції, таким як: фоновий рівень флуоресценції хлорофілу F_O інтенсивність флуоресценції хлорофілу при відкритих реакційних центрах (РЦ) фотосистеми II (ФС II); Z_m –

максимальна флуоресценція при закритих РЦ; F_V – варіабельна флуоресценція, її рівень с індикатором фотохімічних окислювально-відновних процесів, характеризує активність початкових стадій фотосинтезу, розраховується, як різниця між F_m та F_o; параметр F_V / F_m ефективний засіб моніторингу стресових чинників на рослину, оскільки є чутливим до інгібірування світлової фази фотосинтезу; R_{fd} – індекс життєздатності рослини, або, параметр спаду флуоресценції; Area, площа над кривою ІФХ між F_o та F_m; даний параметр пропорційний розміру пула акцепторів електронів Q_A на відновній стороні ФС II і може бути використаний в якості маркеру змін в формі кінетики ІФХ між F_o та F_m; які можуть бути не очевидні з інших параметрів. Параметр розраховується за формулою:

$$Area = \int_{t_{F_o}}^{t_{F_m}} (F_m - F_t)dt, \qquad (1.1)$$

де t_{F_0}, t_{F_m} – час вимірювання параметрів F_o та F_m відповідно.

Експерименти проведено за допомогою портативного приладу «Флоратест» (спектральний діапазон вимірювання інтенсивності флуоресценції від 670 до 770 Нм), який розроблено та підготовлено до серійного виробництва в Інституті кібернетики.

1.2. Нова інформаційна технологія експрес-оцінювання стану рослин в умовах дії стресових факторів

У країнах СНГ кожний рік розвивається аграрна сфера, але не всі фермера можуть дозволити собі постійно використовувати високі технології, що суттєво допомагають збільшити врожай та продуктивність господарства. Далеко не всі

ферми мають свої лабораторії, якісні датчики аналізу ґрунту або безпілотну техніку для якісного аналізу всього посіву рослин на інтервалі її росту, тож з'являється все більше компаній, які надають подібні услуги. Наприклад, нещодавно в Україну прийшли безпілотні дрони для обприскування та десикації посіву, які мають багато переваг перед звичними способами внеску гербіциду з використанням самохідної обприскувальної техніки або літальних пілотних апаратів. Дрони витрачають в декілька разів менший об’єм рідини, не витрачають пальне та не топчуть марно посів, тож все більше аграрії намагаються використовувати новітні технології для покращення всього процесу росту культур.

Після посіву культури зазвичай вносять добриво та потім обприскують гербіцидами щоб уникнути ураження рослин різними шкідливими комахами та бур'яном, але не всі ведуть якісний звіт проведених робіт, тож треба якісно аналізувати зміни після внеску гербіциду та дізнатися яка доза рідини була використана при обприскуванні.

Технології розробки гербіцидів також змінюються з метою покращення їх дії та знаходження більш якісного та корисного составу рідини. Для цього потрібен якісний аналіз впливу состава рідини на рослину, але щоб якісно оцінити гербіцид технологам потрібно зібрати велику кількість даних ураховуючи різноманітні умови росту рослини, клімату, составу ґрунту ітд. Багато систем аналізу почали переносити на безпілотні дрони та використовувати супутники з потужними системами спостерігання за полем.

Однією з основних задач у сучасному промисловому землеробстві та екологічному моніторингу є експресне оцінювання стану рослин в умовах дії стресових факторів, їх стійкості до несприятливих умов навколишнього природного середовища. Це дає змогу своєчасно вжити необхідні агротехнічні заходи для збереження врожаю або зелених насаджень. Найбільш вірогідне оцінювання стійкості рослин до стресових факторів як природного, так і техногенного походження дає прямий метод, який застосовується у польових умовах. Але у польових умовах, як правило, неможливо відтворити стресові умови навантаження на рослину, які рік у рік носять змінний характер. Це не дає змогу фор-

мувати оптимальні стресові умови випробувань рослини на стійкість і, як наслідок, отримувати кількісні показники цієї стійкості. У зв'язку з тим на сьогодні для групової діагностики рослин на стійкість до дії різноманітних стресових факторів застосовують чисельні непрямі лабораторні методи, які мають різну чутливість, достовірність, продуктивність, трудомісткість і тривалість, а також потребують забезпеченості інструментальними засобами, реактивами, кваліфікованим персоналом. До стресових факторів, що впливають на стан рослинного покриву на великих територіях, відносяться мороз, посуха, спека, засолення і підвищена кислотність ґрунту, внесення добрив, біодобавок, пестицидів і гербіцидів, викиди шкідливих елементів в атмосферу та ін.

Для оцінювання дії цих стресових факторів на стан рослин застосовується велика кількість різних засобів і методів. Так, наприклад, для оцінювання впливу морозу на стан рослини використовують метод прямого заморожування, методи, основані на вимірі електропровідності тканин рослини або на вимірі проникності мембран клітини. Для оцінювання впливу посухи та спеки на стан рослини використовують різні фізіологічні та біохімічні методи.

Для оцінювання стійкості рослини до засолення ґрунту створюють штучне засолене середовище, далі саджають рослину, і досліджують її біохімічні та фізіологічні зміни.

Для оцінювання кислотостійкості рослини порівнюють стан рослин у нормальних умовах зі станом рослин у середовищі з різним ступенем кислотності ґрунту.

Методи оцінювання газостійкості рослин можна поділити на біологічні, морфологічні, цитологічні, морфометричні, екологічні, фізіологічні, біохімічні та біофізичні.

Наведені вище методи і методики оцінки стану рослин, по-перше, є, як правило, довготривалими, а не експресними, мають невисоку продуктивність, потребують різних технічних засобів, методичного забезпечення та лабораторного обладнання, для їх використання необхідний кваліфікований персонал. Тому розробка універсальних експресних методів оцінювання стану рослин в

умовах дії стресових факторів є важливою задачею як промислового сільськогосподарського виробництва, так і захисту довкілля на великих площах лісопаркових зон.

Одним з таких методів оцінювання стану живої рослини, на нашу думку, є метод індукції флуоресценції хлорофілу.

Суть методу полягає в наступному. Молекула хлорофілу живої рослини, поглинаючи квант світла, переводить електрон з основного у збуджений стан. Повернення цієї молекули в основний стан відбувається трьома шляхами:

– енергія збудження передається електронно-транспортним ланцюгом сусідній молекулі і так далі, допоки ця енергія не дістанеться реакційного центру фотосинтезу, де вона накопичується у світловому циклі, а у темновому циклі витрачається на фотохімічні реакції;

– електрон повертається на основний рівень, а його енергія виділяється у вигляді тепла;

– електрон повертається на основний рівень, а його енергія випромінюється у вигляді кванту світла або кванту флуоресценції.

Процеси накопичення енергії у листі живої рослини і її випромінювання у вигляді флуоресценції є конкурентними. За інтенсивністю флуоресценції та її змінами можна оцінювати стан рослини в умовах дії стресових факторів незалежно від природи їх походження.

Це пояснюється наступним. Накопичення енергії світла відбувається завдяки фотосинтетичному електронно-транспортному ланцюгу клітини живої непошкодженої стресом рослини. Під дію стресового фактора будь-якої природи перш за все підпадає мембрана, яка є природним бар’єром клітин живої рослини, що відразу викликає каскад зсувів в обміні речовин усієї клітини і перш за все призводить до руйнування електронно-транспортних ланцюгів. Наслідком цих негативних процесів є зміна інтенсивності флуоресценції аж до її зникнення. Таким чином, за результатами реєстрації і подальшої комп’ютерної обробки флуоресценції хлорофілу у живій рослині в умовах дії стресових факторів різної природи можна оперативно визначити її стан. Це дає змогу викорис-

товувати метод індукції флуоресценції хлорофілу як універсальний для оцінювання стану рослини в умовах дії стресових факторів різної природи.

До цього часу метод індукції флуоресценції хлорофілу використовувався у автономних приладах, так званих флуорометрах, для оцінювання стану рослин переважно у лабораторних умовах або на невеликих ділянках у полі. Прикладом такого флуорометру є розроблений в Інституті кібернетики і доведений до серійного виробництва портативний прилад «Флоратест» (рис. 1.1).

Рисунок 1.1 – Портативний прилад "Флоратест"

Використання автономних флуорометрів на великих площах потребує значної кількості цих досить коштовних приладів і відповідного обслуговуючого персоналу. Крім того, результати реєстрації індукції флуоресценції хлорофілу автономним приладом у вигляді кривих мають невисоку наочність, що ускладнює оцінювання стану рослин цим методом. Тому автономні флуорометри, як правило, використовуються в умовах лабораторних досліджень, а в польових умовах мають обмежене застосування.

Авторами статті в рамках міжнародного проекту УНТЦ № 6064 «Розробка і підготовка до серійного виробництва розподілених інтелектуальних біосен-

сорів для захисту довкілля» розроблені і доведені до дослідної експлуатації бездротові інтелектуальні біосенсори для реєстрації індукції флуоресценції хлорофілу, які можуть бути об’єднані у мережі і на відміну від автономних флуорометрів здатні оцінювати стан рослин на великих територіях.

Слід зауважити, що біосенсор з радіоканалом може використовуватися у двох режимах роботи. Перший режим роботи передбачає, що біосенсор працює автономно і через радіоканал передає дані на віддалений блок збору інформації, наприклад, планшетний комп'ютер, ноутбук або спеціально розроблений реєстратор. Другий варіант передбачає, що велика кількість біосенсорів з радіоканалом розміщується на досліджуваній території. При цьому такі бездротові біосенсори самостійно організовують мережу і працюють як єдина бездротова сенсорна мережа.

Запропонована бездротова сенсорна мережа може використовуватися у різних областях прецизійного землеробства та екологічного моніторингу. При цьому вона являє собою мережу з великої кількості бездротових біосенсорів, які об'єднані між собою радіоканалом. Площа покриття подібної мережі може складати від декількох квадратних метрів до декількох квадратних кілометрів за рахунок здатності ретранслювати дані від одного елемента мережі до іншого.

Основною перевагою пропонованої бездротової сенсорної мережі є здатність контролювати в реальному часі стан сільськогосподарських рослин чи зелених насаджень на великих територіях. При цьому деякі вузли мережі можуть вийти з ладу із-за збоїв енергопостачання, фізичних пошкоджень або стороннього втручання. Але відмова одного вузла не впливатиме на роботу усієї мережі. Така відмовостійкість забезпечується розробкою та правильним застосуванням відповідних протоколів передачі даних, алгоритмів функціонування та мережевої взаємодії. Оскільки запропонована бездротова сенсорна мережа працює в умовах реального оточуючого середовища, то елементи мережі мають суттєву стійкість до впливу кліматичних умов. Оскільки вузли мережі часто відмовляють із-за розряджання батареї або впливу фізичних факторів, тому структурі мережі притаманні часті зміни топології після розгортання самої мережі.

Зміна топології зумовлює зміну характеристик самих вузлів: положення, доступність (із-за завад, шумів, рухомих перешкод і т. д.), рівень заряду батареї, неполадки, зміна поставлених задач або ролі в мережі. При виході з ладу будь-якого окремого елемента мережі алгоритми самоорганізації мережі забезпечують створення нових шляхів передачі даних в обхід ділянки мережі, яка вийшла з ладу.

Вузли, як правило, розташовані випадковим чином на всій території спостережень. Кожний з них може здійснювати збір даних і визначати маршрут передачі в центральний вузол до кінцевого користувача. В створеній мережі можна виділити два типи пристроїв. Перший тип – це повнофункціональний вузол. Він може слугувати координатором окремих ланок мережі, так і окремим звичайним вузлом мережі. Такий вузол реалізує загальну модель зв’язку, яка дозволяє “спілкуватись” з іншими вузлами мережі, в такому випадку він є координатором мережі. Другий тип пристроїв – вузли з спрощеними функціями. Тобто це прості вузли з малим ресурсом і вимогами до мережі. Переважно такі вузли можуть зв’язуватися повнофункціональними пристроями і не можуть слугувати координаторами мереж. Основною функцією таких вузлів є вимірювання, попереднє оброблення, стиснення та передавання даних до сусіднього вузла.

Для цілей презиційного землеробства або екологічного моніторингу вузли запропонованої мережі можуть розташовуватися одним з двох способів. Перший спосіб передбачає, що вимірювальні вузли мережі розташовуються за наперед заданим алгоритмом і покривають досліджувану територію рівномірно, наприклад, квадратом n*n вузлів. При цьому щільність вузлів можна контролювати при розробці алгоритму розміщення вузлів. Другий спосіб передбачає, що вузли досліджуваній ділянці розташовуються випадковим чином на різних відстанях один від одного і з різною щільністю на різних ділянках досліджуваної території. При такому способі розташування вимірювальних вузлів головний принцип полягає у тому, що між вузлами має бути стабільний і надійний радіозв'язок.

Для обміну інформацією між вузлами обрано бездротову технологію, яка є доступною у більшості країн. На фізичному рівні протокол пропонує три смуги частот для роботи бездротової сенсорної мережі, зокрема, один канал у смузі частот 868 МГц, десять каналів у смузі частот 915 МГц і 16 каналів у смузі частот 2,4 ГГц для промислових та медичних потреб.

Для реалізації елементів бездротової мережі вибрано модулі бездротової передачі даних зарубіжного виробництва та протокол організації мережевої взаємодії. Мікроконтролер вузла містить 32-бітний процесор з тактовою частотою 32Мгц, модуль безпровідного зв'язку, сумісний із стандартом вбудований 4-канальний 10-бітний АЦП, аналогові та цифрові входи виходи тощо. Цей мікроконтролер за своїми характеристиками повністю відповідає вимогам щодо реалізації робочого вузла мережі, який забезпечує збір, зберігання і передачу даних по бездротовому каналу.

Не слід забувати про вимоги щодо уніфікації або, іншими словами, стандартизації протоколів та елементів, які застосовувалися при створенні мережі. Як, правило, певну стандартизацію у цю область вносить стандарт , який визначає особливості побудови мереж з невисокою пропускною здатністю.

Дані, отримані за результатами вимірів індукції флуросценції хлорофілу, можуть безпосередньо зчитуватися та оброблятися мобільними комп'ютерними платформами, включно безпілотними літальними апаратами (рис. 1.2), або за допомогою хмарних технологій. Для цього в Інституті кібернетики імені В.М.Глушкова НАН України розроблений математичний апарат, який можна використовувати як у хмарних технологіях, так і у мобільних платформах. Можливості цього апарату було перевірено експериментально.

Рисунок 1.2 – Безпілотний літальний апарат для отримання даних

В Інституті кібернетики імені В.М.Глушкова НАН України виконані дослідження впливу дії гербіциду на індукцію флуросценції хлорофілу живої рослини. Нами використовувався промисловий гербіцид «Раундап» і рослини дурману звичайного, які були поділені на групи. Одна з них – контрольна, яка гербіцидом не оброблялася, і дві дослідні групи, що оброблялися різними дозами гербіциду. Для обробки даних цих досліджень щодо визначення дії гербіциду «Раундап» на стан рослини у часі використано апарат нейронних мереж і статистичний аналіз. Для розпізнання стану росли за формою кривої індукції флуросценції хлорофілу використано двошарову нейронну мережу з прямим поширенням сигналів (feed-forward network). Ця нейрона мережа успішно використовується у задачах класифікації. Мережа містила 89 входів та 3 виходи, оскільки

кожна оцифрована крива індукції флуросценції хлорофілу містила 90 відліків (перший відлік нами відкидався як недостовірний). Останній вихідний шар складався з трьох нейронів (три варіанти оцифрованих кривих). За результатами обробки даних апаратом нейронних мереж вже на сьомий день була зафіксована різниця між контрольними і дослідними рослинами, тобто було визначено, що рослина дурману почала руйнуватися під дією гербіциду. Водночас обробка даних зафіксувала різницю між контрольними і дослідними рослинами рослинами тільки на тринадцятий день. Таким чином, методом індукції флуросценції хлорофілу можна визначити дію гербіциду «Раундап» на бур'ян на ранній стадії (через декілька днів), а візуально для цього гербіциду проявляється не раніше, ніж через 15-20 діб. Це дає змогу зберегти рослини від додаткової обробки гербіцидами, тобто зменшити навантаження шкідливими речовинами на навколишнє природне середовище.

Застосовано апарат нейронних мереж оцінювання стану рослин методом індукції флуросценції хлорофілу при визначенні водного дефіциту, оцінювання реакції рослин на посуху, мороз, внесення добрив, виявлення впливу важких металів на стан рослин.

Таким чином доведено, що апарат нейронних мереж придатний для створення промислових методик для оцінювання стану сільськогосподарських рослин, зеленого покриву мегаполісів і лісопарків на великих територіях методом індукції флуросценції хлорофілу, з використанням бездротових сенсорних мереж.

1.3 Змістовна та формальна постановка задачі

Під предметом дослідження ми розглядаємо біологічні ряд, тож розглянемо їх детальніше. Нам потрібно знайти значимі ознаки у ряді, розробити та навчити якісний класифікатор і розпізнати певний тип полива рослин по показнику індукції флуросценції хлорофілу (ІФХ) для подальшого обприскування рослин аграріями та вилучення корисної інформації для подальшого аналізу.

На вхід подається дата сет часових рядів, спостережень ІФХ за певний час випробувань.

Для аналізу зібраних спостережень потрібно побудувати якісний класифікатор, який дасть змогу отримати корисну інформацію з великого набору спостережень та вдосконалити процес десикація культур. Отриманий класифікатор можна буде використовувати та вдосконалити у реальному житті після тестування на великих вибірках з різними факторами середи.

Для реалізації мети системи використовуються інформаційні та технологічні ресурси.

В формальному виді постановка задачі класифікації має наступний вигляд. Нехай $x_i \in X,\ i = \overline{1,n}$, – множина об'єктів ознак, входів моделі, а $y_i \in Y,\ i = \overline{1,n}$, – множина об'єктів відповідей, виходів моделі. Пара $(x_i, y_i) \in X \times Y$ – це прецедент або розмічений об'єкт. Кінцева множина $\{x_i\}, i = \overline{1,n}$, – це така матриця $\{x_{i,j}\},\ i = \overline{1,n},\ j = \overline{1,m},$ розміром $n \times m$, де рядок матриці – це масив ознак одного об'єкта, $\{y_i\}, i = \overline{1,n}$, – вектор відповідей, який складається із значення номеру класу. Пара векторів $\{x_i\}, i = \overline{1,n}$, та $\{y_i\}, i = \overline{1,n}$, називається навчальною вибіркою. Задача класифікації полягає у визначенні функції залежності $f : X \to Y$, яка прогнозує відповіді $y \in Y$ по $x \in X$, іншими словами, потрібно побудувати алгоритм здатний класифікувати довільний об'єкт до одного класу $x \in X$.

1.4 Постановка задачі дослідження

Мета роботи полягає в побудові класифікатору часових рядів, на прикладі біологічних рядів, а саме показнику ІФХ. Потрібно виявити неявні ознаки з набору даних, обрати та побудувати класифікатор на основі обраних дата сетів. Роботу можна поділити на наступні кроки:

– провести аналіз поставленої задачі та описати її формально;

– проаналізувати та обрати підходящі алгоритми класифікаторів;

– знайти найкращу програмну мову для поставленої задачі машинного навчання та розібрати можливі реалізації;

– обрати набор даних та привести його до робочого стану, тобто відформувати дата сет;

– реалізувати класифікатор програмно та наочно вивести результати;

– зробити аналіз отриманих даних та висновки.

2 ВИБІР ТА ОБҐРУНТУВАННЯ МЕТОДУ РОЗВ'ЯЗАННЯ

Класифікація даних – це одна із найвідоміших задач в якісному аналізі даних. Рішенням завдання буде побудований класифікатор часових рядів, який буде навчатися на тестовому наборі даних та потім буде здатним класифікувати нові робочі ряди самостійно. Тобто ми отримаємо класифікатор з принципом навчання з учителем. В процесі навчання класифікатор буде аналізувати вхідні дані та будувати свій алгоритм класифікацій, тобто в нашому випадку – дерево прийняття рішень. Треба обрати найкращий алгоритм та реалізувати його математично. Після навчання отриманий механізм буде здатний найкращим чином знайти значимі ознаки із вхідних часових рядів та встановити належність до того чи іншого типу ряду.

2.1 Модель дерева прийняття рішень

Згідно з найбільш загальним визначенням, дерево прийняття рішень – це ефективний інструмент інтелектуального аналізу даних та передбачуваної аналітики. Він широко застосовується у вирішенні завдань із класифікації та регресії. Алгоритм дерев вперше розробили Адель Катлер та Лео Брейман майже 40 років назад.

За структурою кореневе дерево прийняття рішень до його «прототипу» з живої природи, складається з «гілок» і «листя», як дерево з вузлами, які помічені ознаками, а ребра(гілки), які виходять з вузла відповідають на відповідне питання (ознаку) вершини. Вузол, котрий не має потомків, тобто наступних вузлів називають листком та являє собою остаточний прогноз, рішення задачі класифікації.

Дерево рішень є ієрархічною деревоподібною структурою, що складається з правила виду «Якщо …, то ...» наведено у рис. 2.1. Спуск по дереву від кореня до кінцевого листа дає відповідь дерева на належності до класу. За раху-

нок навчальної множини правила генеруються автоматично в процесі навчання.

На відміну від нейронних мереж, дерева як аналітичні моделі простіше, тому що правила генеруються природною мовою: наприклад, «Якщо реклама привела 1000 клієнтів, то вона налаштована добре».

Правила генеруються з допомогою узагальнення безлічі окремих спостережень (навчальних прикладів), які описують предметну область. Тому їх називають індуктивними правилами, а процес навчання – індукцією дерев рішень.

У навчальній множині для прикладів має бути задане цільове значення, оскільки дерева рішень – моделі, створювані з урахуванням навчання з учителем. За типом змінної виділяють два типи дерев:

– дерево класифікації, у якого цільова змінна дискретна;

– дерево регресії, у якого цільова змінна безперервна.

Насамперед алгоритм дерев використовують в багатьох сферах для рішення задач класифікації. Механізм полягає в розбитті даних, по властивостям доки не будуть отримані однорідні їх підмножини. Отримане потім сукупність правил, дають розбиття, за яким навчений алгоритм буде спроможній проводити класифікацію нових масивів даних власноруч.

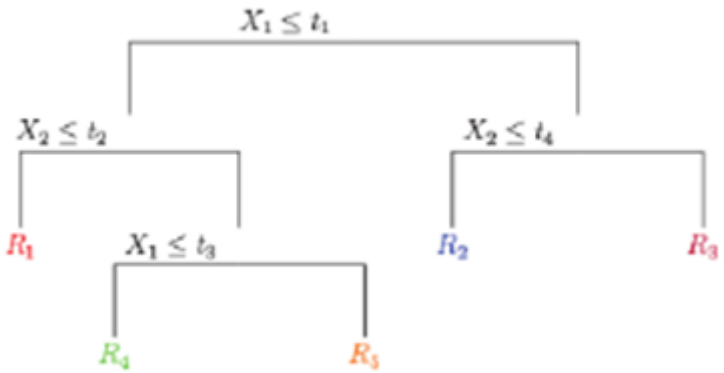


Рисунок 2.1 – Структура дерева прийняття рішень

Дерева рішень – це важливіший компонент випадкового лісу. Дерева здатні підганяти складні набори даних, дозволяючи досліднику побачити, як було прийнято рішення.

Часовий ряд x можна представити у вигляді кінцевої впорядкованої послідовності чисел:

$$x = [x(1), \ldots, x(t)]. \tag{2.1}$$

Вирішальне дерево розбиває весь простір об'єктів на J областей, $R_1, ..., R_J$ при цьому в кожній області дерево повертає константну відповідь $b_1, ..., b_J$. Тож вирішальне дерево можна записати у вигляді наступної суми:

$$b(x) = \sum_{j=1}^{J} [x \in R_j] b_J, \tag{2.2}$$

де x – об’єкт, що розглядається;

b_J – ймовірність належності об’єкту, що розглядається, до J-го класу з області R_J.

Щоб побудувати дерева алгоритм використає признаки якості розбиття, перший з яких – це ентропія Шенона. Ентропія є ступнею «хаоса» системи. При більшої ентропії, система буде гірш впорядкована:

$$S = -\sum_{i=1}^{n} p_i \log_2 p_i, \tag{2.3}$$

де p_i – ймовірність знаходження системи в i-му стані.

Зменшення ентропії Шенона називають приростом інформації та рахують наступним чином:

$$IG(Q) = S_0 - \sum_{i=1}^{m} \frac{n_i}{n} S_i, \tag{2.4}$$

де Q – ознака;

S_i – ентропія в i-му вузлі;

m – кількість груп розбиття;

n_i – кількість елементів вибірки, у яких ознака Q має i-е значення.

Другий критерій – це невизначеність Джині. Невизначеність мінімізує ймовірність помилки при класифікації та має наступний вигляд:

$$G = 1 - \sum_{i=1}^{n} p_i^2. \tag{2.5}$$

Третій критерій – це помилка класифікації, яка допомагає у підрізанні дерева:

$$E = 1 - \max_i p_i. \tag{2.6}$$

За замовчуванням ми використовуємо невизначеність Джині.

Далі нам треба вказати значення глибини для нашого дерева. Глибина дозволяє уникнути побудування хаотичного дерева в різні сторони. Хаотичний зріст приводить до перенавчання та зменшує якість моделі.

2.2 Модель випадкового лісу

Машинне навчання за допомогою розробки деяких алгоритмів автоматичного «навчання» комп’ютера аналізує наявні дані, щоб отримати приховані правила,ознаки, і використовувати ці закони для прогнозування та аналізу невідомих даних. З швидким розвиток епохи мобільного Інтернету, генерування масивних даних і вдосконалення промисловості для розрахування вимоги швидкості та вартості, традиційному мейнфрейму було важко задовольнити потре-

би промислового сектор. В результаті виникає технологія розподілених обчислень. Технології розподілених обчислень шляхом декомпозиції задачі, яке можна вирішити лише за допомогою величезної обчислювальної потужності, декомпозиція на багато підзавдань, а потім призначити ці підзавдання багатьом комп'ютерним вузлам для обробки, і зрештою підсумувати результати розрахунку для отримання кінцевого результату.

Розбиття задачі виконується за принципом побудови «розділяй і володарюй». Основна ідея даного підходу полягає в послідовному розбиття задачі на дрібніші під завдання з метою об'єднання отриманих результатів для прийняття остаточного рішення щодо класифікації зразка. Зазвичай розбиття відбувається до моменту, коли підзадачі нижчого рівня не стають елементарними, або достатньо детальними.

Моделі дерев рішень, нестійкі: навіть невелика зміна в навчальній множині може привезти до істотних змін в структурі всього дерева. В такому випадку корисно використовувати ансамблі моделей.

Одним з перших і відомих видів ансамблів є алгоритм Bagging, особливість полягає в його елементарних класифікаторів, які навчаються і працюють незалежно один від одного. Особливість в тому, що класифікатори не виправляють помилки один одного, а компенсують, враховують їх при голосуванні.

Найяскравіший приклад реалізації беггінгу – це Випадковий Ліс, але на відміну від його основної версії має кілька особливостей, а саме, використовує всередині себе ансамбль регресійних або класифікующих дерев рішень, а ознаки та об'єкти обираються випадковим чином.

Випадковий ліс складається з купи дерев рішень і вихідні категорії визначаються способом класифікації дерева рішень. При побудові єдиного дерева рішень, алгоритм випадкового лісу використовує два процеси випадкового відбору: перший – це випадковий відбір навчальних вибірок, а другий – це випадковий вибір атрибутів характеристик вибірки. Після побудови всіх дерева рішень, остаточний результат класифікації визначається методом рівноваги.

Алгоритм лісу на початку роботи вилучає випадкову бутстрап-вибірку

фіксованого розміру *n*.

Бутстрап-вибірка – це набір даних, що отримується в результаті генерації повторних вибірок з вибірки, що досліджується, з метою апроксимації вибіркового розподілу статистики.

Далі будується стартове дерево прийняття рішень за допомогою отриманої вибірки. В кожному вузлі стартового дерева виконуються наступні кроки:

– обирається число ознак без повернення $d = \sqrt{m}$;

– відповідно до цільової функції обирається найкраща ознака для розщеплення вузла;

– попередні два кроку повторяються стільки раз, скільки ми задали дерев у системі;

– визначається мітка класу завдяки агрегування прогнозу з кожного дерева чином мажоритарного голосування, тобто на основі більшості голосів.

2.3 Адаптивний бустінг як алгоритм посилення класифікатору

Дерева рішень добре підходить до задач класифікації та їх використовують не тільки у випадкових лісах. Далі мі розглянемо алгоритм, який розробили для покращення якості ансамблів. Бустінг не тільки поєднує слабкі класифікатори, а й покращує їх роботу.

Більше 30 років тому двоє британських Кернс та Веліант розробили бустінг, коли шукали відповідь на питання: «Чи може набір слабких навчальних алгоритмів створити сильний навчальний алгоритм?».

Посилення(підвищення) гіпотези, тобто бустінг – це техніка ансамблю, при якої нові класифікатори з’являються послідовно та виправляють помилки попередніх моделей. В бустінгу моделями можуть виступами як простіші алгоритми, так і потужні нейронні мережі. Нові моделі додаються доки можливо робити вдосконалення класифікаторів.

У машинному навчанні бустінг підвищує потужності слабких класифіка-

торів, для перетворення їх на сильні. Під слабким класифікатором слід розуміти любий алгоритм, здатний класифікувати об’єкт трохи краще ніж при випадковому вгадуванні, тобто з невеликою точністю. Також слабкі алгоритми легко обчислюються, тож при їх поєднанні отримаємо сильний ансамбль класифікаторів, при цьому підвищуючи точність.

Популярні приклади – алгоритм адаптивного бустінгу (AdaBoost), який зважує точки даних, які важко передбачити та градієнтний бустінг (GradientBoosting).

Градієнтний бустінг – це підхід, коли створюються нові моделі, які передбачають залишки або помилки попередніх моделей, а потім додаються разом для остаточного прогнозування. Його називають збільшенням градієнта, оскільки він використовує алгоритм спуску градієнта для мінімізації втрат при додаванні нових моделей.

Цей підхід підтримує як регресію, так і проблеми класифікації прогнозного моделювання, він намагається підігнати новий класифікатор до залишкових помилок, які були допущені попереднім класифікатором. Наприклад, для рішення задачі регресії існує реалізація AdaBoostRegressor, яка в основі використовує дерева регресії.

Алгоритм AdaBoost можна використовувати для підвищення продуктивності будь-якого алгоритму машинного навчання.

AdaBoost означає "Adaptive Boosting" або адаптивний бустінг. Наприклад, AdaBoost можна використовувати для розпізнавання облич, оскільки він є стандартним алгоритмом для таких завдань.

Рівняння для класифікації може виглядати так:

$$F(x) = sign(\sum_{m=1}^{M} \theta_m f_m(x)), \tag{2.7}$$

де f_m – це m-ий слабкий класифікатор, де m відповідає за відповідну йому вагу.

AdaBoost найкраще працює зі слабкими навчальними алгоритмами, а

найбільш поширеними алгоритмами, які використовуються з AdaBoost, є одно-рівневі дерева рішень. Розглянемо алгоритм детальніше. Якщо у нас є набір даних, в якому *n* – кількість точок та $x_i \in R^d, y_i \in \{-1,1\}$ Де –1 буде негативним класом, а 1 – позитивним. Тоді вага кожної точки буде ініціалізована, як показано нижче:

$$w(x_i, y_i) = \frac{1}{n}, i = 1,...,n\,. \tag{2.8}$$

Кожне m у наступному виразі ми змінюватимемося від 1 до M. Для початку потрібно вибрати слабкий класифікатор із найменшою виваженою помилкою класифікації, застосувавши класифікатор до набору даних.

$$\varepsilon_m = E_{w_m}[1_{y \neq f(x)}]. \tag{2.9}$$

Потім порахуємо вагу m-ого слабкого класифікатора, як показано нижче:

$$\theta_m = \frac{1}{2}\ln\left(\frac{1-\varepsilon_m}{\varepsilon_m}\right). \tag{2.10}$$

Вага додатна для будь-якого класифікатора з точністю вище 50%. Чим більша вага, тим точніше класифікатор. Вага стає негативною, коли точність падає нижче 50%. Передбачення можна поєднувати, інвертуючи знак. Таким чином класифікатор з точністю 40% можна перетворити на класифікатор з точністю 60%. Так класифікатор вноситиме внесок у підсумкове передбачення, навіть якщо він працював гірше, ніж випадкове вгадування. Однак остаточний результат не зміниться під впливом класифікатора, точність якого дорівнює 50%.
Експонента в чисельнику завжди буде більше 1 у разі неправильної класифікації із класифікатора з позитивною вагою. Після ітерації вага неправильно кла-

сифікованих об'єктів зросте. Класифікатори з негативною вагою поведуться аналогічним чином. Тут різниця в інверсії знака: правильна класифікація стане неправильною. Остаточний прогноз можна розрахувати шляхом урахування вкладу кожного класифікатора та обчислення суми їх виважених прогнозів.
Вага для кожної точки оновлюватиметься таким чином:

$$w_{m+1}(x_i, y_i) = \frac{w_m(x_i, y_i)\exp[-\theta_m y_i f_m(x_i)]}{Z_m}, \tag{2.11}$$

де Z_m – це нормалізуючий параметр. Він потрібний, щоб переконатися, що сума всіх ваг екземпляра дорівнює 1.

AdaBoost можна використовувати для розпізнавання облич, оскільки він є стандартним алгоритмом для таких завдань. Він використовує каскад відбраковування, що складається з кількох шарів класифікаторів. Коли область розпізнавання не виявляє обличчя на жодному шарі, вона відбраковується. Перший класифікатор області відкидає негативну область, щоб звести вартість обчислень до мінімуму. Незважаючи на те, що AdaBoost використовується для поєднання слабких класифікаторів, принципи AdaBoost також використовуються для пошуку кращих ознак кожного шару в каскаді.

AdaBoost допомагає вибрати навчальний набір кожного класифікатора, який навчається на основі результатів роботи попереднього класифікатора. Що ж до об'єднання результатів, то алгоритм визначає, яку вагу потрібно надати кожному класифікатору залежно від отриманої відповіді. Він поєднує слабкі класифікатори для створення сильного та виправлення помилок класифікації, а також є вкрай успішним алгоритмом бустінгу для задач двійкової класифікації.

2.4 Метрики оцінки роботи класифікатору

Для наших класифікаторів можна використовувати оціночні метрики, які дадуть нам змогу для якісної оцінки результатів. Далі для розбору метрик ми повинні розібрати термін confusion matrix.

Наприклад, для розбору матриці в нас буде два кінцевих класи та алгоритм класифікації. Тоді матриця помилок буде виглядати як наведено на рис. 2.2.

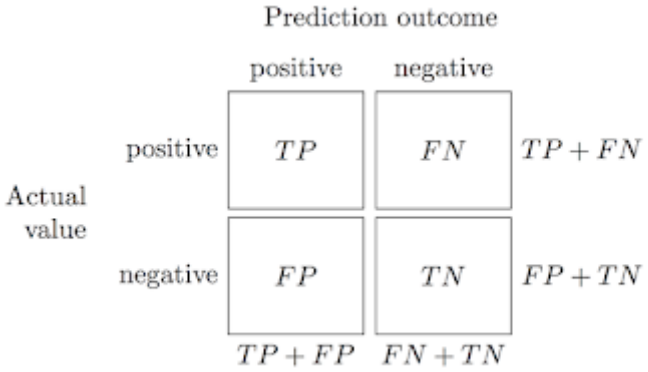


Рисунок 2.2 – Confusion Matrix

Розглянемо рядки, Actual value показує відповідь нашого алгоритму. Далі по вертикалі маємо prediction – істину відповідь на прикладі, що розглядаємо. Таким чином можемо отримати два види помилок False Positive (FP) та False Negative (FN).

Почнемо з метрики precision, яка відповідає на таке питання: яка частина позитивних відповідей є вірно ідентифікована або яка частина з позитивних відповідей є насправді позитивними? Розрахувати метрику можемо використовуючи наступну формулу:

$$PRE = \frac{TP}{TP + FP}.$$

Наступна метрика це recall або повнота, вона відповідає на питання: яка частина насправді позитивних зразків була ідентифікована вірно? Розраховуємо наступним чином:

$$REC = \frac{TP}{TP + FN}.$$

Для отримання певного компромісу між двома розглянутими метриками (precision і recall) ми розглянемо їх поєднання, метрика F1 є середнім гармонічним точності та повноти та змінюється від 0 до 1. Має таку формулу:

$$F1 = 2\frac{PRE \times REC}{PRE + REC}.$$

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Машинне навчання та вибір мови реалізації

Машинне навчання – це галузь штучного інтелекту (ШІ) та інформатики, яка зосереджується на використанні даних і алгоритмів для імітації того, як люди навчаються, поступово підвищуючи точність його рішень.

Машинне навчання є важливим компонентом зростаючої галузі науки про дані. Завдяки використанню статистичних методів алгоритми навчаються робити класифікації або прогнози, відкриваючи ключові ідеї в рамках проектів аналізу даних. Ці ідеї згодом сприяють прийняттю рішень у додатках і підприємствах, ідеально впливаючи на ключові показники зростання.

Технологічні розробки, пов’язані з накопиченням і процесорною потужністю, дозволяють створити деякі інноваційні продукти, які ми знаємо і любимо сьогодні, наприклад, рекомендаційні двигуни у медіа порталів або самокеровані автомобілі.

В сучасному світі майже в кожній галузі можна зчитати для аналізу будь-який процес. В основі цієї революції лежать інструменти та методи інтелектуального аналізу які керують цим, від обробки величезної купи даних, що генеруються щодня для навчання моделей до здійснення корисних дій. Глибокі нейронні мережі, досягнення в класичному машинному навчанні і масштабованість GPU загального призначення, стали критичними компонентами штучного інтелекту, що дозволяють робити багато приголомшливих проривів та знижають складність рішення складних задач. Python продовжує бути найулюбленішою мовою для наукових обчислень та машинного навчання, при цьому покращуючи продуктивність і потужність, дозволяючи використовувати низькорівневі бібліотеки та чисті API високого рівня.

Вважається, що найкраща реалізація любого інтелекту пишеться на мові Python. Python являє собою мову високого рівня програмування, яку можна використовувати для рішення безлічі різних задач починаючи з різних наук про

дані і до внутрішньої веб-розробки або авто-тестування.

Засновником була компанія Python Foundation на початку 1990-х і вже стає потужнішим інструментом для аналізу, класифікації та передбачення даних, широко використовується в обробці великих даних.

Особливу підтримку мові надає численне співтовариство розробників машинного навчання, яке зосереджено на швидко зростаючому напрямку штучного інтелекту. Завдяки активної спільноті для Python з'явилося безліч готових бібліотек для машинного навчання. Це мова переносних незалежний, тому її можна адаптувати практично до будь-якої операційної системи.

За рахунок простого синтаксису, достатку навчальних матеріалів і високої швидкості виконання коду Python дозволяє всі зусилля спрямувати безпосередньо на машинне навчання. Код реалізації пишеться швидко. Один з головних мінусів є складність відслідковування помилок в коді, це пов'язано з розростанням кодової бази, відповідно, з її складністю. У деяких випадках перевірка коду може виливатися в значні фінансові витрати, віднімати багато часу і позначатися на продуктивності проекту.

У висновках зазначимо, що Python практично безальтернативний варіант для машинного навчання. Для рішення задач навчання штучного інтелекту, у пітона немає недоліків. Код пишеться легко, документації на готові фреймворки написані якісно, що полегшують написання коду.

3.2 Вибір матеріалів та підготовка даних

Дані були зібрані за допомогою прибору «Флоратест», який встановили на безпілотний дрон та проводили спостереження через фіксований інтервал часу. Об’єктом дослідження обрано рослину соя (лат. *Glycine max*), засухостійкий сорт Сіверка. Дослідження виконано в квітні-травні 2016 року. Рослини вирощені в 9 горщиках в кімнатних умовах з насіння. Проведено 12 серій вимірювань. Рослини поділено на три групи за поливом.

Таблиця 3.1 – Схема експерименту

група V1	з поливом, доза поливу рівна 50 мл води на 1 кг ґрунту
група V2	з поливом, доза поливу рівна 150 мл води на 1 кг ґрунту
група V3	без поливу

Після першої серії вимірювання кривих індукції флуоресценції хлорофілу припинено полив у групі V3, цей день вважаємо першим днем експерименту.

Таблиця 3.2 – План експерименту

Номер вимірювання	Дата
01	18.04.16
Припинено полив групі V3	
02	19.04.16
03	20.04.16
04	22.04.16
05	25.04.16
06	26.04.16
07	27.04.16
08	29.04.16
09	04.05.16
10	06.05.16
11	10.05.16
12	13.05.16

Щоб наочно порівняти часові ряди індукції флуоресценції хлорофілу ми виводимо усередненні ряди кожних класів на одну площину на рис. 3.1.

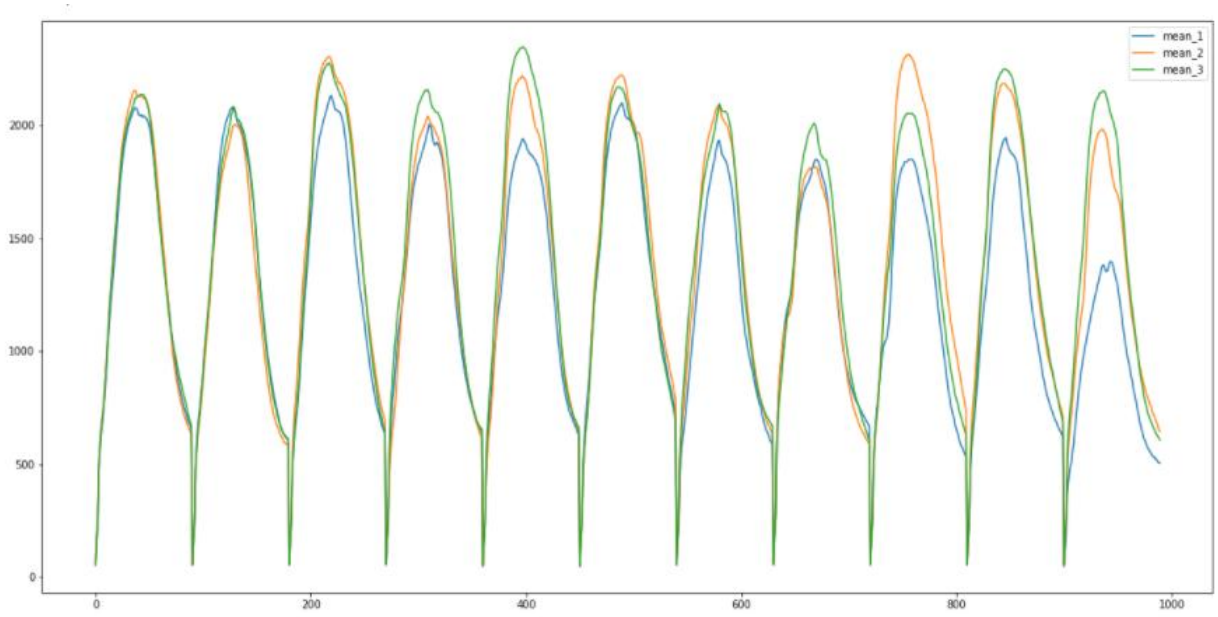


Рисунок 3.1 – Імпортовані бібліотеки Python

3.3 Опис програми

В опису програми ми розглянемо всю реалізацію наших класифікаторів та детально розглянемо кожний блок коду.

На початку потрібно імпортувати та об’явити все необхідні в роботі бібліотеки.

Бібліотеки time та datetime допоможе в роботі з часовим рядами, які потрібно відформувати одним чином, не втрачаючи дату спостережень та для формування виводу.

Нам потрібна буде бібліотека Pandas для читання нашої бази даних з файлу та у форматі CSV (coma separated values) для подальшої роботи з даними.

Далі при розбору дата сету нам потрібно вилучити корисні ознаки із рядів, в цьому допоможе бібліотека tsfresh. З цим ми зможемо автоматично згенерувати корисні ознаки для процесу навчання наших ансамблів.

Реалізації основних ансамблів містяться у бібліотеці sklearn, тож ми імпортуємо звідси RandomForestClassifier, AdaBoostClassifier та DecisionTreeClassi-

fier щоб визначити базовий класифікатор ансамблю AdaBoost. Також імпортуємо матрицю невідповідностей та звіт класифікації. На останок добавимо інструмент роботи з графіками, для наочності рядів. На рис. 3.2 бачимо всі розглянуті бібліотеки та модулі Python.

```
•[1]: %matplotlib inline
import csv
import time
import pandas
import datetime
from tsfresh import extract_features
from tsfresh.feature_extraction.settings import TimeBasedFCParameters
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.metrics import plot_confusion_matrix
from sklearn.ensemble import RandomForestClassifier, AdaBoostClassifier
from sklearn.tree import DecisionTreeClassifier
from matplotlib import pyplot as plt
from matplotlib.pylab import rcParams

rcParams['figure.figsize'] = 20,10
```

Рисунок 3.2 – Імпортовані бібліотеки Python

На початку треба обробити дата сет, підготувати до роботи. Усі початкові таблиці були переформовані в один csv файл. До файлу ми добавляємо нову колонку, яка слугує ідентифікатором для алгоритму вилучення ознак. Після обробки зчитуємо дата сет у об’єкт на рис. 3.3.

```
[2]: with open('mean_ds.csv') as inp, open('mean_otp.csv', 'w') as out:
    reader = csv.reader(inp)
    writer = csv.writer(out)
    writer.writerow(['id'] + next(reader))
    writer.writerows([i] + row for i, row in enumerate(reader, 1))
data_set = 'mean_otp.csv'
df = pandas.read_csv(data_set)
df.head()
```

[2]:

	id	exp_day	date	measurement	group	mean
75	76	1	2016-04-18	26	1	1845.333333
76	77	1	2016-04-18	26	2	1920.000000
77	78	1	2016-04-18	26	3	1888.000000
78	79	1	2016-04-18	27	1	1882.666667
79	80	1	2016-04-18	27	2	1952.000000

Рисунок 3.3 – Обробка та зчитування даних

Проблемою при побудові моделі класифікатора з якою ми зіткнемося це

вилучення важливих ознак із часового ряду для детального опису даних щоб далі добре навчити моделі. За допомогою функції extract_features нам вдалося визначити достатньо значимих ознак. Важливо отримати багато значимих ознак із ряду, щоб отримати максимально якісний ансамбль. На наступному рис. 3.4 відбувається вилучення ознак.

Ознаки ряду можуть бути досить різними. Починаючи з тривіальних значень максимуму, мінімуму або середнього значення до куту нахилу ряду, медіани та тренду на відрізку графіка.

```
[4]: settings_time = TimeBasedFCParameters()
     settings_time

[4]: {'linear_trend_timewise': [{'attr': 'pvalue'},
       {'attr': 'rvalue'},
       {'attr': 'intercept'},
       {'attr': 'slope'},
       {'attr': 'stderr'}]}

[5]: df.info()
     X_tsfresh = extract_features(
         df, column_id="id", column_value='mean', column_sort='date', default_fc_parameters=settings_time)

     <class 'pandas.core.frame.DataFrame'>
     Int64Index: 627 entries, 75 to 2831
     Data columns (total 6 columns):
      #   Column       Non-Null Count  Dtype
     ---  ------       --------------  -----
      0   id           627 non-null    int64
      1   exp_day      627 non-null    int64
      2   date         627 non-null    object
      3   measurement  627 non-null    int64
      4   group        627 non-null    int64
      5   mean         627 non-null    float64
     dtypes: float64(1), int64(4), object(1)
     memory usage: 34.3+ KB
     Feature Extraction: 100%|██████████| 20/20 [00:51<00:00,  2.59s/it]
```

Рисунок 3.4 – Пошук ознак

Для передачі сету ансамблеві ми попередньо видаляємо колонку з міткою належності до класу та на перестраховуємось від використання нульових даних, видаляючи пусті строки на рис. 3.5.

```
X_tsfresh.dropna(how='any', inplace=True)
X_tsfresh = df.drop(columns=['group', 'date'])
X_tsfresh
```

Рисунок 3.5 – Чистка Обробка та читання даних

Далі на рис. 3.6 поділяємо вхідні дані на два набори. Перший для процесу

навчання, що складає 85% від усіх даних та другий тестовий набір. Тестові дані містять лише 15% зразків. Всього спостережень було не так багато, тож зразків не достатньо щоб виділити їх більше на процес тестування.

```
X_train, X_test, y_train, y_test = train_test_split(X_tsfresh, y, test_size=0.20)
```

```
X_train
```

	id	exp_day	measurement	mean
383	384	2	38	2064.000000
898	899	5	30	1904.000000
1204	1205	8	42	2181.333333
352	353	2	28	1765.333333
2787	2788	26	30	1248.000000
...	...	...	...	...
1171	1172	8	31	2170.666667
1696	1697	10	26	1813.333333
1431	1432	9	28	1925.333333
2557	2558	23	43	2138.666667
401	402	2	44	2026.666667

Рисунок 3.6 – Розбиття зразків на набори

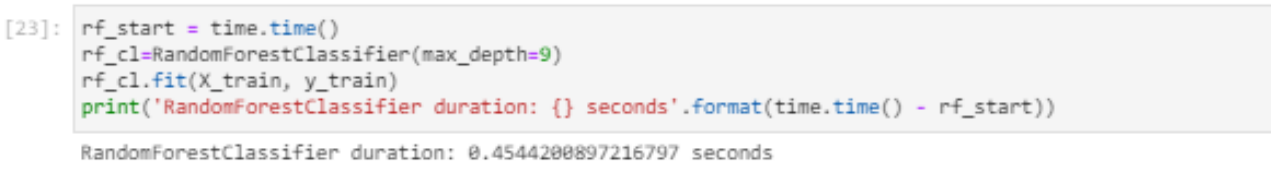
Далі на рис. 3.7 ми починаємо реалізовувати ансамблі. Перший ансамбль – це адаптивний бустінг. Ми обрали відому реалізацію AdaBoostClassifier, через багату документацію та якісну роботу. Щоб засікти час, котрий ансамбль використовує на навчання ми зчитали роботу в секундах за допомогою методу time(). Стандартний конструктор ансамблю потребує визначення базового алгоритму класифікації, тож ми присвоюємо ансамблю дерево прийняття рішень як base_estimator. Глибину побудови дерев ми обрали 9, це оптимальне значення, що дозволяє використати всі ознаки та уникнути хаосу у структурі дерев.

```
[14]: ada_start = time.time()
      cl = AdaBoostClassifier(base_estimator=DecisionTreeClassifier(max_depth=9))
      cl.fit(X_train, y_train)
      print('AdaBoostClassifier duration: {} seconds'.format(time.time() - ada_start))
```

```
AdaBoostClassifier duration: 0.15690016746520996 seconds
```

Рисунок 3.7 – Визначення ансамблю адаптивного бустінгу

Останній важливий блок коду на рис. 3.8 описує реалізацію ансамблю беггінгу, а саме реалізацію випадкового лісу. На відміну від адаптивного бустінгу, випадковий ліс вже базується на деревах прийняття рішень, тож нам треба визначити тільки їх глибину, яку ми теж залишаємо рівною 9. Тим самим чином ми замірили час роботи ансамблю та бачимо, що випадковий ліс навчався втричі довше ніж бустінг. Тож в швидкості побудови та навчання перевагу має адаптивний бустінг.



```
[23]: rf_start = time.time()
      rf_cl=RandomForestClassifier(max_depth=9)
      rf_cl.fit(X_train, y_train)
      print('RandomForestClassifier duration: {} seconds'.format(time.time() - rf_start))

      RandomForestClassifier duration: 0.4544200897216797 seconds
```

Рисунок 3.8 – Визначення ансамблю випадкового лісу

4 РЕЗУЛЬТАТИ ОБЧИСЛЮВАЛЬНОГО ЕКСПЕРИМЕНТУ

Щоб оцінити якість класифікації ми викликаємо функцію plot_confusion_matrix на рис. 4.1. Розглянемо матрицю детальніше, діагональні елементи представляють кількість точок, для яких передбачена мітка дорівнює істинній мітці, тоді як недіагональні елементи – це ті, які неправильно позначені класифікатором. Чим вище значення діагоналі матриці плутанини, тим краще, що вказує на багато правильних передбачень.

На рисунках показано матрицю плутанини з нормалізацією. Цей вид нормалізації може бути цікавим у разі дисбалансу класів, щоб мати більш наочну інтерпретацію того, який клас неправильно класифікується.

```
plot_confusion_matrix(cl, X_test, y_test, normalize='true')
```

Рисунок 4.1 – Виклик plot_confusion_matrix

Далі для повного звіту викликаємо classification_report на рис. 4.2. Звіт класифікації використовується для вимірювання якості прогнозів за алгоритмом класифікації. Скільки передбачень правдивих, а скільки хибних. Точніше, для прогнозування показників звіту про класифікацію, як показано нижче, використовуються істинно позитивні, хибнопозитивні, справді негативні та помилкові негативи. Виведений звіт класифікації відображає нормалізовані у відсотковому відношенні показники точності, відкликання, F1 для моделі у вигляді таблиці. що полегшує наочну оцінку моделі ансамблю в різних метриках.

```
print(classification_report(y_test, cl.predict(X_test)))
```

Рисунок 4.2 – Виклик методу classification_report

Матриця невідповідностей в купі з повним звітом класифікації дають нам наочне представлення точності роботи класифікаторів, тож можемо оцінити та порівняти разом дві ансамблеві техніки.

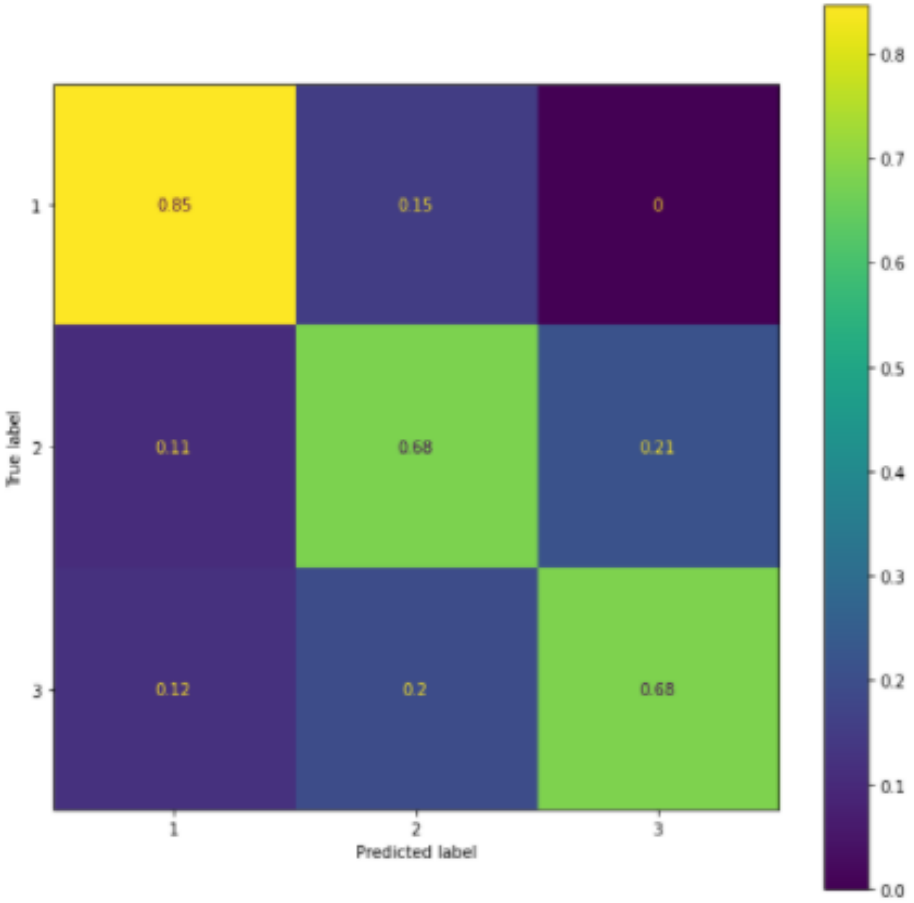


Рисунок 4.3 – Матриця невідповідностей ансамблю RandomForestClassifier

```
print(classification_report(y_test, rf_cl.predict(X_test)))
```

	precision	recall	f1-score	support
1	0.73	0.85	0.79	26
2	0.61	0.68	0.64	28
3	0.82	0.68	0.75	41
accuracy			0.73	95
macro avg	0.72	0.74	0.73	95
weighted avg	0.74	0.73	0.73	95

Рисунок 4.4 – Результати класифікації ансамблю RandomForestClassifier

Тепер розглянемо результати класифікатору AdaBoostClassifier. Можемо зразу побачити, що приріст точності достатньо великий, та більший за беггінг. З найліпшої точністю маємо перший клас, тобто групу рослин з поливом в 50 мл. Найменшу точність класифікатор дав для другого класу, тобто групи з поливом потрійною дозою в 150 мл. Результати наведені на рис. 4.5, 4.4.

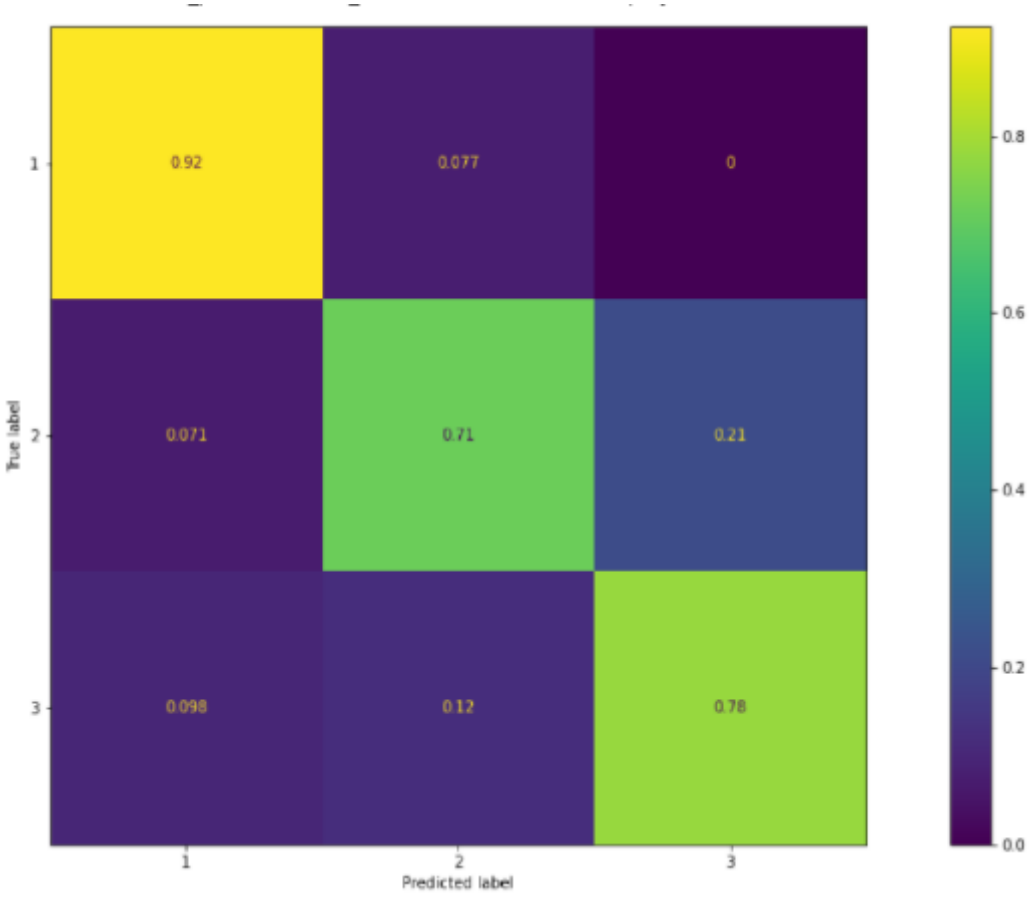


Рисунок 4.5 – Матриця невідповідностей ансамблю AdaBoostClassifier

```
print(classification_report(y_test, cl.predict(X_test)))
```

	precision	recall	f1-score	support
1	0.80	0.92	0.86	26
2	0.74	0.71	0.73	28
3	0.84	0.78	0.81	41
accuracy			0.80	95
macro avg	0.79	0.81	0.80	95
weighted avg	0.80	0.80	0.80	95

Рисунок 4.6 – Результати класифікації ансамблю AdaBoostClassifier

Для наочного висновку зберемо результати точності ансамблів в єдину таблицю 4.1 та проаналізуємо. Наглядно видно, що реалізація бустінгу має кращі результати по всім класам.

Менш точні результати ніж у бустінгу нам дав випадковий ліс. Стабільно на кожному класі маємо прибавку у точності класифікації у адаптивного бусті-

нгу. У додаток швидкість навчання бустінгу була вище у три рази, що дає значну перевагу над випадковим лісом.

Таблиця 4.1 – Порівняння точності беггінгу та бустінгу

Алгоритм	Random Forest Classifier			Adaptive Boost Classifier		
Метрика	precision	recall	f1	precision	recall	f1
1 клас	0.73	0.85	0.79	0.80	0.92	0.86
2 клас	0.61	0.68	0.64	0.71	0.71	0.73
3 клас	0.82	0.68	0.75	0.84	0.78	0.81

5 АНАЛІЗ МОЖЛИВИХ ЗАСТОСУВАНЬ

В межах кваліфікаційної роботи була проведено дослідження, яке є одним з способів класифікації біологічних часових рядів.

Показано, що на сьогодні для оцінювання стану рослин на великих територіях під дією стресових факторів різної природи існує багато лабораторних методів і методик, які потребують досить коштовного обладнання, реактивів і кваліфікованого персоналу.

В результаті виконання міжнародного проекту експериментально доведено, що замість багатьох методів і методик для оцінювання стану рослин на великих територіях можна використовувати універсальний метод індукції флуросценції хлорофілу, який має велику чутливість до дії на рослини стресових факторів різної природи.

Для промислового використання цього методу в Інституті кібернетики імені В.М. Глушкова НАН України розроблена інформаційна технологія, яка включає бездротові сенсори і мережі на їх основі для одночасного збору даних про стан рослин на великих територіях сільськогосподарських угідь, зеленого покриву мегаполісів і лісопарків, і ефективний математичний апарат обробки цих даних, придатний для використання у хмарному середовищі або мобільних обчислювальних платформах. Після збору даних треба якісно провести їх аналіз та вилучити корисну для аграріїв інформацію.

Остаточний класифікатор добре підходить під задачу багато класової класифікації, завдяки потужної композиції із декількох слабких класифікаторів, дерев прийняття рішень. Обрані біологічні часові ряди мають дуже схожі показники для кожного класу, що ускладнює задачу та не дає змогу отримати відмінну точність.

Для порівняння ми взяли дві популярні композиції класифікаторів із беггінгу та бустінгу, а саме Random Forest Classifier та Adaptive Boost Classifier. Проведенні експерименти та наглядне порівняння показало, що адаптивний бу-

стінг показав ліпші результати точності ніж були у випадкового лісу, тож ми можемо рекомендувати цей алгоритм для рішення подібних задач.

Отриманий класифікатор може зіграти важливу роль у моніторингу полів сельхоз призначення та при аналізу зібраних даних за допомогою безпілотного дрона із бездротовим біосенсором для реєстрації індукції флуоресценції хлорофілу. Також результати даної атестаційної роботи можуть бути застосовані при виборі методу класифікації часових рядів в інших біологічних або фізичних, економічних, соціальних системах і т. д з метою вилучення корисної інформації з неструктурованого набору даних.

ВИСНОВКИ

У межах даної кваліфікаційної роботи розглядалась задача побудови композиції класифікатору часових рядів на основі біологічних рядів.

Було обрано дві основні техніки ансамблевих класифікаторів та знайдено найкращі реалізації на мові Python. Ця мова була обрана, як найліпша мова вирішення задач машинного навчання, через багату документацію, легкість кодування та великій вибір готових бібліотек. На початку було сформульовано постановку задачі машинного навчання та проаналізовано її актуальність. Класифікація проходила на основі біологічних часових рядів, а саме індексу флуоресценції хлорофілу, який дає змогу найбільш швидким та інформативним шляхом визначити вплив зовнішніх факторів на рослину та є найкращим методом екологічного моніторингу.

Було знайдено та використано корисну бібліотеку для пошуку ознак із часових рядів, щоб ансамблі могли якісно навчитися на тренувальних дата сетах.

Процес реалізації показав нам, що кількість вхідних даних та різниці у класах має велику значимість. Складно якісно навчити ансамбль на невеликому наборі даних. Також через схожість усіх зразків класифікатору треба мати достатньо ознак розпізнання ознак та належності до класу. Глибина побудови дерев прийняття рішень теж мала важливу роль, треба обирати оптимальну глибину щоб уникнути перенавчання або втрати важливих вузлів дерева. Нарешті було знайдено оптимальну реалізацію ансамблів, яка дала найліпші результати для нашого невеликого набору даних. Адаптивний бустінг виконав класифікацію краще, ніж випадковий ліс, та приніс великі показники точності, у середньому 80%.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Логунова А. В. Phyton и Машинное обучение. Москва : “ДМК Издательство”, 2017. 418 с.

2. Knowledge Base. URL : https://www.it.ua/ru/knowledge-base/technology-innovation/ (дата звернення: 05.05.2020).

3. Aurelien G. Hands-On Machine Learning with Scikit-Learn & TensorFlow. O`REILLY Inc, 2017. 750 p.

4. Tsay R. Analysic of financial time series. John Wiley & Sons, 2005. 543 p.

5. OTUS Tsfresh. URL : https://otus.ru/nest/post/1024/ (дата звернення: 22.05.2020).

6. U.S. BUREAU OF LABOR STATISTICS CPI. URL : https://www.bls.gov/cpi/questions-and-answers.htm#Question_3 (дата звернення: 07.05.2020).

7. Breiman L. Bagging predictors in Machine Learning. 1996. 145 p.

8. Breiman L. Random Forests in Machine Learning. 2001. 254 p.

9. Jason B. Deep Learning for Natural Language Processing. Jason Brownlee. 2017. 397 p.

10. Romanov V., Brayko Yu, Galelyuka I., Imanutdinova R., Kytayev O., Palagin O., Sarakhan Ye., Starodub M., Fedak V, Portable Biosensor: from Idea to Market//International Journal Information Theories & Applications. 2021. Vol. 19, No. 2, p. 126-131.

11. Груша В. М. Обробка результатів експериментальних досліджень, проведених з використанням портативного флуорометра «Флоратест»//Комп’ютерні засоби, мережі та системи. 2015. с. 109-116.

12. Рубин А. А. Биофизические методы в экологическом мониторинге//Соросовский оброзовательный журнал, 2000. No. 2, с. 197-210.