



Основи Data Science

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалавр)
Галузь знань	G Інженерія, виробництво та будівництво
Спеціальність	G3 Електрична інженерія
Освітня програма	Інжиніринг інтелектуальних електротехнічних та мехатронних комплексів
Статус дисципліни	Вибіркова
Форма навчання	Денна/заочна
Рік підготовки, семестр	2 курс, осінній семестр
Обсяг дисципліни	4 кредити / 120 годин
Семестровий контроль/ контрольні заходи	Залік, МКР
Розклад занять	http://roz.kpi.ua/
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: доцент, к.т.н. Кулаковський Леонід Ярославович, тел. 097-453-65-46, email: l.kulakovskiy@kpi.ua Практичні: асистент Хотян Артем Анатолійович
Розміщення курсу	https://classroom.google.com/c/Nzc1MjM4OTk1NDA5?cjc=i5zpxjd2

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Силабус освітнього компонента «Основи Data Science» складено відповідно до освітньої програми підготовки бакалаврів «Інжиніринг інтелектуальних електротехнічних та мехатронних комплексів» галузі знань G3 Електрична інженерія.

Метою вивчення дисципліни є формування у студентів здатностей:

- застосовувати базові методи та інструменти аналізу даних для розв'язання прикладних інженерних задач в електротехнічних комплексах та мехатронних системах;
- здійснювати попередню обробку, візуалізацію та інтерпретацію даних із використанням сучасних програмних засобів;
- використовувати основи машинного навчання для побудови моделей прогнозування, класифікації та оптимізації в інтелектуальних системах.

та закріплення у студентів наступних компетентностей:

(ФК 10) усвідомлення необхідності постійно розширювати власні знання про нові технології в електроенергетиці, електротехніці та електромеханіці.

(ФК 14) здатність вирішувати комплексні спеціалізовані задачі і практичні проблеми, пов'язані з розробкою автоматичних систем управління із врахуванням експертного досвіду.

(ФК 15) здатність застосовувати методи теорії автоматичного керування, системного аналізу та числових методів для розроблення математичних моделей електротехнічних та мехатронних комплексів, аналізу якості їх функціонування із використанням новітніх комп'ютерних технологій.

Предметом навчальної дисципліни є основи збору, обробки, аналізу та моделювання даних для підтримки прийняття рішень в електротехнічних та мехатронних комплексах.

Програмні результати навчання, на формування та покращення яких спрямована дисципліна: (ПРН06) Застосовувати прикладне програмне забезпечення, мікроконтролери та мікропроцесорну техніку для вирішення практичних проблем в професійній діяльності; (ПРН08) обирати і застосовувати придатні методи для аналізу і синтезу електромеханічних та електроенергетичних систем із заданими показниками; (ПРН19) вміти самостійно вчитись, опановувати нові знання і вдосконалювати навички роботи з сучасним обладнанням, вимірювальною технікою та прикладним програмним забезпеченням; (ПРН21) застосовувати сучасні методи оптимізації при синтезі електротехнічних та мехатронних систем та комплексів

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Для успішного засвоєння дисципліни студент повинен володіти освітніми компонентами "Вища математика", "Основи програмування".

Знання та уміння, одержані в процесі вивчення кредитного модуля «Основи Data Science» є необхідними для кожного фахівця даної спеціальності, які вирішують інженерні завдання у сфері електротехніки та при вивченні таких дисциплін: «Інтелектуальні системи керування електротехнічними комплексами», «Моделювання електротехнічних систем», «Машинне навчання», «Нелінійні задачі та ідентифікація електротехнічних та мехатронних систем» тощо.

3. Зміст навчальної дисципліни

Тема 1. Введення до вивчення дисципліни.

Тема 1.1. Data Science як галузь знань для аналізу, обробки та інтерпретації великих обсягів даних.

Тема 1.2. Python як інструмент аналітика даних. Інтерактивне середовище розробки Jupyter Notebook.

Розділ 2. Основні синтаксису мови Python для аналізу та обробки даних.

Тема 2.1. Змінні, цілі числа, рядки, числа з плаваючою комою, перетворення типів.

Тема 2.2. Структури даних.

Тема 2.3. Оператори циклу, імпортери, функції. Функціональне програмування та спискові вирази в Python.

Розділ 3. Бібліотека NumPy для роботи з багатовимірними масивами та матрицями.

Тема 3.1. Основи роботи з бібліотекою NumPy.

Тема 3.2. Особливості роботи з масивами в NumPy.

Розділ 4. Обробка табличних даних за допомогою бібліотеки Pandas.

Тема 4.1. Основи роботи з Pandas: DataFrame та Series.

Тема 4.2. Очищення та підготовка даних у Pandas.

Тема 4.3. Перетворення та збагачення даних у Pandas.

Розділ 5. Візуалізація даних у Data Science:

Тема 5.1. Візуалізація даних у Data Science: Основи роботи з бібліотекою Matplotlib.

Тема 5.2. Аналітична візуалізація в Data Science: розширені можливості бібліотек Matplotlib, Plotly, Seaborn.

Тема 5.3. Інтерактивна візуалізація та аналіз часових рядів засобами Plotly.

Тема 6.1. Основи статистики в бібліотеці SciPy для Data Science.

Тема 6.2. SciPy для обробки та аналізу даних.

Тема 6.3. SciPy для моделювання та оптимізації в Data Science.

4. Навчальні матеріали та ресурси

Базова

1. Яковенко А. В. Основи програмування. Python. Частина 1 : навч. посіб. / А. В. Яковенко. — Київ : КПІ ім. Ігоря Сікорського, 2021. — 120 с.
2. Сілен Д., Мейсман К., Алі М. Основи Data Science і Big Data. Python і наука про дані / Д. Сілен, К. Мейсман, М. Алі ; пер. з англ. — Київ : Наш Формат, 2022. — 384 с.
3. Маккінні В. Python для аналізу даних. Data Wrangling з Pandas, NumPy та Jupyter / Вес Маккінні ; пер. з англ. — Харків : Видавництво «Фабула», 2023. — 544 с.4.
4. Методичні вказівки «Використання Python, бібліотек NumPy та Matplotlib» / НТУ «ХПІ». — Харків : НТУ «ХПІ», 2021. — 48 с.
5. Беррі П. Head First. Python / Пол Беррі ; пер. з англ. — Київ : Yakaboo Publishing, 2023. — 320 с.
6. Васильєв О. Програмування мовою Python. Алгоритмічні структури і стратегії. Теорія та практика / О. Васильєв. — Київ : Yakaboo Publishing, 2023. — 320 с.
7. Підручник з Python — Python 3.13.7 documentation / Python Software Foundation. — Електронний ресурс. — 2025.
8. Горобець О. Ю., Горобець С. В., Хахно К. Ю. Мова Python для інженерних та наукових задач : підручник. — Київ : КПІ ім. Ігоря Сікорського, 2024.
9. Основи Data Science. Практикум [Електронний ресурс]: навч. посіб. для здобувачів ступеня бакалавра за освітньою програмою «Інжиніринг інтелектуальних електротехнічних та мехатронних комплексів» спец. G3 Електрична інженерія / Л. Я. Кулаковський; КПІ ім. Ігоря Сікорського. — Електрон. текст. дані (1 файл). — Київ : КПІ ім. Ігоря Сікорського, 2026. —170 с.

Допоміжна

1. Методичні вказівки до лабораторної роботи «Основи роботи з бібліотекою NumPy» з курсу «Обробка даних Python» / уклад. : С. М. Коваленко, С. В. Коваленко, О. В. Шматко. — Харків : НТУ «ХПІ», 2021. — 48 с.
2. Інтелектуальний аналіз даних та машинне навчання. Частина 1. Базові методи та засоби аналізу даних / Я. В. Іванчук, В. І. Месюра, А. А. Яровий, О. Д. Манжілевський – Вінниця : ВНТУ, 2021. — 69 с.
3. Методичні вказівки до лабораторної роботи «Основи роботи в середовищі Jupyter Notebook» з курсу «Обробка даних Python» [Електронний ресурс / уклад.: С. М. Коваленко, С. В. Коваленко, О. В. Шматко ; Нац. техн. ун-т "Харків. політехн. ін-т". — Електрон. текст. дані. — Харків, 2021. — 28 с.
4. Методичні вказівки до лабораторної роботи «Основи роботи з бібліотекою Matplotlib» з курсу «Обробка даних Python» / уклад. : С. М. Коваленко, С. В. Коваленко, О. В. Шматко. — Харків : НТУ «ХПІ», 2021. — 28 с.
5. Miller C. Hands-On Data Analysis with NumPy and pandas / Curtis Miller. — Packt Publishing, 2019. — 330 p.
6. Nelli F. Python Data Analytics: Data Analysis and Science Using Pandas, Matplotlib, and the Python Programming Language / Fabio Nelli. — Springer, 2015. — 442 p.
7. Klein B. Master Data Analysis with Python: NumPy, Matplotlib, and Pandas / Bernd Klein. — Python Course, 2021. — 312 p.
8. Сліпченко В. Задачі з програмування мовою Python / Валентина Сліпченко. — Київ : Ліра-К, 2021. — 180 с.
9. Імовірнісні основи обробки даних: Збірник тестових завдань [Електронний ресурс]; уклад.: Батіна О.А., Н.Ю. Філіпова. — Електронні текстові данні (1 файл: 3,3 Мбайт). — Київ : КПІ ім. Ігоря Сікорського, 2020. — 153 с.

10. Онлайн тренажер по вивченню мови програмування python та його основних бібліотек
Режим доступу: <https://www.w3schools.com/python/>

11. Онлайн тренажер по вивченню мови програмування python та його основних бібліотек
Режим доступу: <https://www.tutorialspoint.com/>

Літературу, бібліографія якої подана із посиланням, можна знайти в інтернеті. Літературу, бібліографія якої не містить посилання, можна знайти в бібліотеці КПІ ім. Ігоря Сікорського. Обов'язковою для прочитання є базова література [1]-[9]. Усі інші літературні джерела є факультативними, з ними рекомендується ознайомитись.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Застосовуються стратегії активного навчання, які визначаються такими методами та технологіями: методи проблемного навчання (дослідницький метод); особистісно-орієнтовані технології, засновані на таких формах і методах навчання як візуалізація та інформаційнокомунікаційні технології, зокрема електронні презентації для лекційних занять. Викладання проводиться у формі лекцій та практичних занять.

Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на літературу)
1	Лекція 1. Тема 1.1. Data Science як галузь знань для аналізу, обробки та інтерпретації великих обсягів даних. Основна мета освітнього компоненту. Історія виникнення та розвитку цього напрямку. Інструменти та етапи робочого процесу у Data Science. Порівняння особливостей застосування мов програмування R, Python, Matlab для вирішення задач Data Science. Література: [4, 5]
2	Лекція 2. Тема 1.2. Python як інструмент аналітики даних. Дистрибутиви Python. Інтерактивне середовище розробки Jupyter Notebook. Структура блокнота та принципи обчислень. Оформлення блокноту за допомогою Markdown та із застосуванням мови розмітки для створення високоякісних документів Latex для виведення тексту в формі виразів, таблиць, рівнянь тощо. Література: [2, 3, 6]
3	Лекція 3. Тема 2.1. Основи типів даних та змінних у Python. Змінні, цілі числа, рядки, числа з плаваючою комою, перетворення типів. Найменування та використання змінних. Маніпуляції та форматування з числами. Булеві типи даних. Вбудовані типи Boolean та None. Конвертація типів. Література: [2, 3, 6]
4	Лекція 4. Тема 2.2. Структури даних. Списки (list()). Використання списків як стеків. Розуміння списку та вкладених списків. Кортежі та послідовності (tuple()). Набори (set). Заморожений набір (frozensets). Словники. Створення словників. Доступ до даних в словнику, перебір елементів словника. Додавання, видалення даних в словнику. Методи словників Python Література: [4, 5]
5	Лекція 5. Тема 2.3. Оператор циклу for, while. Оператори керування циклами break, continue. Додаткові оператори else. Вкладені (nested) цикли. Input та швидкий огляд імпортів. Консольні програми. Функції. Визначення функцій (використовуючи ключове слово def), виклик функцій, параметри та аргументи, оператор return для повернення значень, вбудовані функції (наприклад, print(), sqrt()), а також функції з користувацьких

	бібліотек (наприклад, <code>math.sqrt()</code>). Функціональне програмування в Python. Елементи функціонального програмування. Спискові вирази: лямбда-вирази (<code>lambda</code>), <code>map()</code>, <code>filter()</code>. Ітератори, спискові включення (<code>list comprehension syntax</code>), замикання (<code>closures</code>), анотації типів Генератори списків/словників та інших колекцій. Література: [3, 5, 6]
6	Лекція 6. Тема 3.1. Основи роботи з бібліотекою NumPy. Стандартні типи даних NumPy. Методи автозаповнення та формування числових діапазонів. Методи створення матриць. Операції над матрицями. Агрегатні функції NumPy. Методи формування масивів. Створення масивів без використання списків. Атрибути <code>ndarray</code>. Доступ до окремих елементів <code>ndarray</code>. Література: [4, 5, 8]
7	Лекція 7. Тема 3.2. Особливості роботи з масивами в NumPy. Властивості та уявлення NumPy -масивів. Операції із масивами. Зрізи масиву: доступ до підмасивів. Використання булевих масивів в якості маски. Просунуте індексування. Переформатування масивів. Індксація, зрізи, ітерування по масивах. Спискова та маскова індксація (<code>fancy & masking indexing</code>). Література: [4, 5]
8	Лекція 8. Тема 3.3. Особливості роботи з масивами в NumPy. Транспонування матриць та векторів. Робота з розмірностями. Базові математичні операції з масивами. Поняття "векторизації". Методи генерації псевдовипадкових чисел та перемішування елементів масиву. Методи математичної статистики. Елементи лінійної алгебри, операції з матрицями та векторами. Транслявання масивів (<code>array broadcasting</code>). Література: [6, 7]
10	Лекція 9. Тема 4.1. Основи роботи з Pandas: <code>DataFrame</code> та <code>Series</code>. Введення до об'єктів Pandas. Розгляд основних можливостей бібліотеки. <code>Series</code> як спеціалізований словник. Побудова об'єктів <code>Series</code>. Побудова об'єктів <code>DataFrame</code>. <code>Index</code> як незмінний масив. <code>Index</code> як множина. Індксація та вибір даних. Вирівнювання даних за мітками індексу. Індксатори: <code>loc</code> і <code>iloc</code>. Вибір даних у <code>DataFrame</code>. Створення <code>DataFrame</code> з різних джерел (словники, списки, CSV, Excel). Операція з даними в Pandas. Відбір значень за допомогою індексу. Ієрархічна індксація. Універсальні функції: збереження індексу. Універсальні функції: Вирівнювання індексу (<code>Index Alignment</code>). Література: [6, 7]
11	Лекція 10. Тема 4.2. Очищення та підготовка даних у Pandas. Проблеми якості даних: пропущені значення, дублікати, шум. Виявлення пропущених значень (<code>isnull()</code>, <code>notnull()</code>). Стратегії обробки пропусків: Видалення (<code>dropna()</code>), Заповнення (<code>fillna()</code>). Виявлення та видалення дублікатів (<code>duplicated()</code>, <code>drop_duplicates()</code>). Зміна типів даних (<code>astype()</code>), обробка дат. Робота з категоріальними даними. Виконання операцій із ковзним вікном. Створення зрізів. Чисельні та статистичні методи. Мова запитів. Графічні можливості бібліотеки Pandas. Стандартизація назв колонок, категорій, форматів. Література: [6, 7]
12	Лекція 11. Тема 4.3. Перетворення та збагачення даних у Pandas. Створення нових колонок на основі існуючих. Застосування функцій до колонок (<code>apply()</code>, <code>map()</code>, <code>lambda</code>). Групування та агрегація (<code>groupby()</code>, <code>agg()</code>). Злиття та об'єднання таблиць (<code>merge()</code>, <code>concat()</code>). Робота з часовими рядами: індксація, ресемплінг. Підготовка даних до машинного навчання: нормалізація, <code>one-hot encoding</code>. Література: [6, 7]
13	Лекція 12. Тема 5.1. Візуалізація даних у Data Science: Основи роботи з Matplotlib. Розширена візуалізація засобаби бібліотеки Matplotlib. Основні компоненти графіка:

	Figure, Axes, Axis. Побудова простих графіків: plot(), scatter(), bar(), hist(). Налаштування графіка. Область рисування (Axes). Візуалізація розподілу, трендів та кореляцій. Налаштування графіків для аналітичного звіту. Підписи, легенди, кольори, стилі. Збереження графіків для презентацій та звітів. Збереження графіків у файли (формати PNG, PDF тощо). Порівняння Matplotlib з іншими бібліотеками (Seaborn, Plotly). Література: [6, 7]
14	Лекція 13. Тема 5.2. Аналітична візуалізація в Data Science: розширені можливості Matplotlib Інтерактивна візуалізація засобами бібліотеки Plotly. Використання стилів (plt.style.use()). Інтерактивність у Jupyter Notebook (магічні команди %matplotlib inline, %matplotlib notebook). Використання стилів (plt.style.use()). Побудова багатопанельних графіків (subplot, subplots) для порівняння даних. Візуалізація часових рядів та трендів. Робота з Pandas DataFrame та NumPy масивами у Matplotlib. Візуалізація результатів аналізу даних засобами бібліотеки Pandas. Бібліотека Seaborn: heatmap, pairplot, boxplot. Література: [6, 7]
14	Лекція 14. Тема 5.3. Інтерактивна візуалізація та аналіз часових рядів засобами Plotly. Основи роботи з бібліотекою Plotly. Створення інтерактивних лінійних, точкових та комбінованих графіків. Робота з часовими рядами та їх представлення у Data Science. Обробка дат і часу, індексація часових даних. Аналіз трендів та сезонних компонентів часових рядів. Використання ковзного середнього (rolling average) для згладжування даних. Інтерактивні можливості Plotly: масштабування, фільтрація, підказки (hover), вибір діапазонів даних. Візуалізація та аналіз параметрів технічних систем у часі. Побудова аналітичних панелей та інтерактивних звітів для підтримки прийняття рішень. Література: [6, 7].
15	Лекція 15. Тема 6.1. Основи статистики для Data Science. Описова статистика: середнє, медіана, мода, дисперсія. Розподіли: нормальний, експоненційний. Візуалізація статистичних характеристик.
16	Лекція 16. Тема 6.2. SciPy для обробки та аналізу даних. Огляд структури SciPy: основні підмодулі (stats, optimize, integrate, signal, interpolate). Статистичний аналіз з scipy.stats. Описова статистика (середнє, медіана, дисперсія). Тестування гіпотез (t-тест, χ^2 -тест, кореляція). Інтерполяція та апроксимація даних (scipy.interpolate). Обробка сигналів та фільтрація (scipy.signal), згладжування.
17	Лекція 17. Тема 6.3. SciPy для моделювання та оптимізації в Data Science. Розв'язання нелінійних рівнянь та систем (scipy.optimize.root, fsolve). Мінімізація функцій (scipy.optimize.minimize) — приклади з Data Science. Підбір параметрів моделі. Фітинг кривих. Чисельне інтегрування (scipy.integrate.quad, odeint) — застосування в моделюванні процесів. Робота з матрицями та лінійною алгеброю (scipy.linalg). Побудова моделей на основі диференціальних рівнянь. Застосування SciPy у задачах машинного навчання (наприклад, оптимізація функції втрат).
18	Лекція 18. Залікове оцінювання

Практичні заняття:

Практичні заняття з дисципліни проводяться викладачем згідно навчального плану. **Основною ціллю** практичних занять є закріплення теоретичних положень і набуття умінь їх практичного застосування шляхом виконання певних відповідно сформульованих завдань.

№ з/п	Назва теми, що виноситься на практичне заняття
Практичне заняття №1	Тема 1.2. Python як інструмент аналітики даних. Інтерактивне середовище розробки Jupyter Notebook. Встановлення на ПК дистриб'ютива Anaconda та останньої версії мови програмування python. Робота в командному рядку CMD (Command Prompt). Базові команди по визначенню та встановленню директорії знаходження python файлів в CMD. Перевірка і встановлення необхідних бібліотек python в CMD. Запуск Jupyter Notebook. Ознайомлення з інтерфейсом Jupyter Notebook. Робота з комірками. Оформлення документації за допомогою Markdown. Заголовки, списки, посилання, вставка зображень, форматування тексту.
Практичне заняття №2	Тема 2.1. Змінні, цілі числа, рядки, числа з плаваючою комою, перетворення типів. Оголошення та використання змінних у Python. Створення змінних різних типів: int, float, str. Перетворення типів даних. Використання функцій int(), float(), str() для конвертації. Маніпуляції з числами та форматування виводу. Арифметичні операції, округлення, форматування через f-strings. Робота з рядками. Конкатенація, методи upper(), lower(), replace(), форматування тексту. Перетворення типів даних. Логічні операції та вирази. Практичні завдання по створенню та конвертації типів даних та форматуванні тексту.
Практичне заняття №3	Тема 2.2. Структури даних. Створення та базові операції зі списками (list). Додавання, видалення елементів, зрізи, ітерація. Основні операції над структурами даних. Використання списків як стеків. Методи append() та pop(). Приклад: моделювання черги команд для керування електроприводом. Кортежі (tuple) та їх застосування. Зберігання незмінних даних, наприклад, номінали елементів. Набори (set) та заморожені набори (frozenset). Практичні завдання включають обчислення середньої напруги для кожного режиму, визначення максимального струму, перевірку перевищення температури (наприклад > 80°C) роботи технічних систем, створення словників, кортежів, множин з аналізом режимів роботи.
Практичне заняття №4	Тема 2.3. Оператори циклу, імпортери, функції. Поняття циклу в програмуванні. Цикли for та while. Ітерація по списках, діапазонах (range()), умовні цикли. Оголошення та використання функцій. Функції в Python. Визначення функцій (def), виклик, параметри, аргументи, return. Використання функцій для обробки даних. Поєднання циклів та функцій. Приклади завдань: реалізувати функцію обчислення середньої напруги та застосувати її до всіх режимів; реалізувати функцію обчислення середнього струму та знайти режим з найбільшим значенням; створити функцію для визначення максимальної температури та знайти аварійний режим; обчислити потужність для кожного режиму за допомогою функції тощо
Практичне заняття №5	Тема 2.3. Обробка даних. Спискові включення та базові методи аналізу Спискові включення (list comprehensions). Порівняння циклів та спискових включень. Вбудовані функції для роботи з даними. Фільтрація та перетворення даних. Lambda-функції. Функції map(), filter(). Порівняння підходів до обробки даних. Приклади завдань: Обчислити список середніх значень напруги для всіх режимів (list comprehension); обчислити список

	потужностей $P=U \cdot I$ для всіх режимів; відібрати всі значення температури, що перевищують 80°C; знайти режим, у якому максимальний струм (використовуючи <code>list comprehension</code>) тощо.
Практичне заняття №6	Тема 3.1. Основи роботи з бібліотекою NumPy. Масиви та обробка даних. Призначення бібліотеки NumPy. Основна структура даних – ndarray. Створення масивів. Індукування масиву. Зрізи (slicing). Векторизація. Агрегатні функції. Робота з випадковими даними. Приклади завдань: Створити масив за допомогою <code>np.arange</code>; створити масив значень від 1 до n; обчислити квадрат елементів; вивести масив у зворотному порядку; виконати піднесення до степеня використовуючи функції бібліотеки Numpy.
Практичне заняття №7	Тема 3.2. Розширені операції над масивами NumPy. Булеві масиви та логічна обробка даних. Створення та використання булевих масивів (масок). Фільтрація даних за умовами. Логічні операції над масивами NumPy: <code>&</code>, <code> </code>, <code>~</code>. Застосування складних логічних умов для аналізу даних. Використання функції <code>np.where()</code> для умовного вибору та заміни значень. Підрахунок кількості елементів, що задовольняють задані умови. Визначення унікальних значень за допомогою <code>np.unique()</code>. Аналіз даних інженерних систем із використанням напруги, струму та температури. Приклади завдань: відбір температур, що перевищують критичне значення; визначення режимів із перевантаженням; заміна аномальних значень; обчислення потужності та визначення критичних режимів роботи системи.
Практичне заняття №8	Тема 3.3. Багатовимірні масиви та матричні обчислення в NumPy. Створення та використання двовимірних масивів (матриць). Основні операції над матрицями та векторами. Матричне множення. Транспонування матриць. Обчислення визначника та знаходження оберненої матриці. Розв'язання систем лінійних рівнянь засобами NumPy. Використання матриць для аналізу параметрів технічних систем. Приклади завдань: формування матриць напруг, струмів і температур; обчислення матриці потужностей; знаходження середніх, мінімальних та максимальних значень; дослідження взаємозв'язків між параметрами системи; аналіз режимів роботи електротехнічних та мехатронних комплексів за допомогою матричних методів.
Практичне заняття №9	Тема 4.1. Основи роботи з бібліотекою Pandas. Табличні дані та DataFrame. Призначення бібліотеки Pandas та її використання для аналізу даних. Структури даних Series та DataFrame. Створення та дослідження структури DataFrame. Доступ до рядків, стовпців та окремих елементів даних. Використання методів <code>loc</code> та <code>iloc</code>. Створення нових параметрів на основі існуючих даних. Фільтрація, сортування та агрегування даних. Використання функцій <code>mean()</code>, <code>max()</code>, <code>min()</code>, <code>sum()</code> для аналізу технічних параметрів. Приклади завдань: створення таблиці режимів роботи електродвигуна; обчислення потужності як нового стовпця; визначення режимів із критичною температурою; сортування за напругою або струмом; знаходження середніх значень параметрів та аналіз ефективності роботи системи.
Практичне заняття №10	Модульна робота
Практичне заняття №11	Тема 4.2. Очищення та підготовка даних у Pandas. Поняття Data Cleaning та його роль у процесі аналізу даних. Робота з пропущеними значеннями (NaN). Використання методів <code>dropna()</code> та <code>fillna()</code>.

	Виявлення та видалення дублікатів. Трансформація даних та створення нових параметрів. Застосування функцій <code>apply()</code> та умовної обробки даних. Фільтрація, сортування та класифікація записів. Приклади завдань: виявлення та заповнення пропущених значень у вимірюваннях; видалення дублікатів; створення нових колонок з розрахунковими параметрами; класифікація режимів роботи обладнання; аналіз даних після очищення та підготовки до подальших досліджень.
Практичне заняття №12	Тема 4.3. Розширені можливості Pandas для аналізу даних. Застосування функцій до стовпців DataFrame за допомогою <code>apply()</code> та <code>map()</code> . Групування даних засобами <code>groupby()</code> та виконання агрегатних обчислень. Об'єднання таблиць за допомогою <code>merge()</code> та <code>concat()</code> . Робота з категоріальними даними. Створення аналітичних звітів на основі декількох джерел інформації. Приклади завдань: обчислення нових показників для кожного запису; групування даних за режимами роботи; обчислення середніх та максимальних значень параметрів; об'єднання таблиць вимірювань і характеристик обладнання; аналіз залежностей між параметрами технічних систем.
Практичне заняття №13	Тема 5.1. Візуалізація даних засобами Matplotlib. Роль візуалізації в аналізі даних. Основи роботи з бібліотекою Matplotlib. Побудова лінійних графіків (Line Plot), точкових діаграм (Scatter Plot), стовпчикових діаграм (Bar Chart) та гістограм (Histogram). Налаштування елементів графіка: заголовків, підписів осей, легенд та сітки. Аналіз отриманих графічних залежностей. Приклади завдань: побудова графіка зміни напруги та струму; аналіз залежності між параметрами системи; порівняння режимів роботи за допомогою стовпчикових діаграм; дослідження розподілу температур за допомогою гістограм та формування висновків щодо стану системи.
Практичне заняття №14	Тема 5.2. Аналітична візуалізація даних засобами Seaborn. Призначення бібліотеки Seaborn та її переваги для статистичного аналізу даних. Побудова діаграм <code>scatterplot</code> , <code>pairplot</code> , <code>heatmap</code> , <code>boxplot</code> та <code>histplot</code> . Аналіз кореляцій між параметрами технічних систем. Дослідження розподілу даних та виявлення аномалій. Інтерпретація результатів статистичної візуалізації. Приклади завдань: побудова кореляційної матриці для параметрів системи; аналіз взаємозв'язку між напругою, струмом та температурою; виявлення викидів за допомогою <code>boxplot</code> ; аналіз розподілу вимірювань та оцінка стабільності роботи обладнання.
Практичне заняття №15	Тема 5.3. Інтерактивна візуалізація та аналіз часових рядів засобами Plotly. Створення інтерактивних графіків із використанням бібліотеки Plotly. Робота з часовими рядами та перетвореннями дат і часу. Побудова інтерактивних лінійних, точкових та комбінованих графіків. Аналіз трендів і екстремальних значень. Використання ковзного середнього для згладжування даних. Інтерактивний аналіз параметрів технічних систем. Приклади завдань: побудова часових графіків зміни напруги та потужності; виявлення пікових навантажень; дослідження довгострокових трендів; побудова комбінованих графіків декількох параметрів та оцінка стабільності режимів роботи системи..
Практичне заняття №16	Тема 6.1. Чисельні методи та оптимізація засобами бібліотеки SciPy. Призначення бібліотеки SciPy для розв'язання інженерних та наукових задач. Чисельне інтегрування функцій. Розв'язання нелінійних рівнянь. Оптимізація

	функцій та пошук екстремумів. Використання засобів лінійної алгебри для роботи з матрицями. Аналіз та інтерпретація результатів чисельних обчислень. Приклади завдань: обчислення інтеграла для визначення енергії або заряду; розв'язання нелінійних рівнянь технічних систем; знаходження оптимальних параметрів роботи обладнання; визначення оберненої матриці та її використання в інженерних розрахунках; аналіз отриманих результатів і оцінка їх фізичного змісту..
Практичне заняття №17	Тема 6.2. Моделювання динамічних систем та розв'язання диференціальних рівнянь. Поняття динамічної системи та математичної моделі. Використання функції <code>odeint()</code> для чисельного розв'язання диференціальних рівнянь. Моделювання перехідних процесів в електротехнічних та мехатронних системах. Аналіз часових характеристик систем. Візуалізація результатів моделювання. Приклади завдань: моделювання заряджання та розряджання RC-ланцюга; дослідження зміни температури в технічній системі; аналіз впливу початкових умов і параметрів моделі на поведінку системи; визначення часу досягнення усталеного режиму та побудова графіків динамічних процесів.
Практичне заняття №18	Тема 6.3. Обробка експериментальних даних та статистичний аналіз у SciPy. Методи інтерполяції, екстраполяції та апроксимації даних. Побудова математичних моделей за експериментальними вимірюваннями. Статистична обробка результатів досліджень. Оцінювання середніх значень, дисперсії та стандартного відхилення. Перевірка статистичних гіпотез та аналіз достовірності результатів. Приклади завдань: інтерполяція значень температури або струму між точками вимірювання; апроксимація експериментальних даних поліноміальними функціями; дослідження впливу шуму на результати вимірювань; статистичне порівняння двох серій експериментів; аналіз трендів і формування висновків щодо роботи технічної системи..

6. Самостійна робота студента

Самостійна робота студента передбачає:
підготовку до аудиторних занять – 38 год;
підготовку до модульної контрольної роботи – 4 год;
підготовку до залікового оцінювання – 6 год.

Контрольна робота

Метою контрольної роботи є закріплення та перевірка теоретичних знань із освітнього компонента, набуття студентами практичних навичок самостійного вирішення задач прийняття рішень.

Модульна контрольна робота (МКР) виконується після вивчення Розділів 1-3 та виконання практичних занять 1-9. Контрольна робота проводиться у середовищі Google Classroom. Кожен студент отримує індивідуальне завдання, відповідно до якого необхідно вирішити оптимізаційну задачу щодо прийняття рішень методом експертного оцінювання або методами лінійного чи нелінійного програмування в залежності від варіанту.

7. Політика навчальної дисципліни (освітнього компонента)

Політика навчальної дисципліни «Основи Data Science» заснована на корпоративній політиці КПІ ім. Ігоря Сікорського.

КПІ ім. Ігоря Сікорського є вільним і автономним центром освіти, що покликаний давати адекватні відповіді на виклики сучасності, плекати й оберігати духовну свободу людини, що робить її спроможною діяти згідно з власним сумлінням; її громадянську свободу, яка є основою формування суспільно відповідальної особистості, та академічну свободу і доброчесність, що є головними рушійними чинниками наукового поступу. Внутрішня атмосфера Університету будується на засадах відкритості, прозорості, гостинності, повазі до особистості.

Вивчення навчальної дисципліни «Основи Data Science» потребує: підготовки до практичних занять, опрацювання рекомендованої основної та додаткової літератури.

Підготовка та участь у практичних заняттях передбачає: ознайомлення з програмою навчальної дисципліни та планами практичних занять; вивчення теоретичного матеріалу; виконання завдань, запропонованих для самостійного опрацювання.

Система вимог, які викладач ставить перед студентом:

- правила відвідування занять: заборонено оцінювати присутність або відсутність здобувача на аудиторному занятті, в тому числі нараховувати заохочувальні або штрафні бали. Відповідно до РСО даної дисципліни бали нараховують за відповідні види навчальної активності на лекційних та практичних заняттях.
- правила поведінки на заняттях: студент має можливість отримувати бали за відповідні види навчальної активності на лекційних та практичних заняттях, передбачені РСО дисципліни. Використання засобів зв'язку для пошуку інформації на гугл-диску викладача, в інтернеті, в дистанційному курсі на платформі Сікорський здійснюється за умови вказівки викладача;
- політика дедлайнів та перескладань: якщо студент не проходив або не з'явився на МКР (без поважної причини), його результат оцінюється у 0 балів. Перескладання результатів МКР не передбачено;
- політика щодо академічної доброчесності: Кодекс честі Національного технічного університету України «Київський політехнічний інститут» <https://kpi.ua/files/honorcode.pdf> встановлює загальні моральні принципи, правила етичної поведінки осіб та передбачає політику академічної доброчесності для осіб, що працюють і навчаються в університеті, якими вони мають керуватись у своїй діяльності, в тому числі при вивченні та складанні контрольних заходів з дисципліни «Основи Data Science»;
- при використанні цифрових засобів зв'язку з викладачем (мобільний зв'язок, електронна пошта, переписка на форумах та у соцмережах тощо) необхідно дотримуватись загальноприйнятих етичних норм, зокрема бути ввічливим та обмежувати спілкування робочим часом викладача.
- здобувач вищої освіти повинен дотримувати навчально-академічної етики та графіка навчального процесу; бути зваженим, уважним.

8. Види контролю та рейтингова система оцінювання результатів навчання (РСО)

Поточний контроль: МКР (модульна контрольна робота) проводиться перед другим календарним контролем на практичному занятті у присутності викладача (32 бали), 17 практичних занять (48 балів за практичне заняття = $17 \times 4 = 68$). МКР виконується у вигляді відповіді на теоретичне запитання з лекційного матеріалу та одну практичну роботу. По закінченню заняття робота над МКР закінчується і не підлягає переписуванню. МКР оцінюється в 32 бали за такими критеріями:

– «відмінно» – повна відповідь на теоретичне питання (не менше 90 % потрібної інформації), надані відповідні обґрунтування та особистий погляд та вирішено правильно задачу – 32-29 балів;

– «добре» – достатньо повна відповідь на теоретичне питання (не менше 75 % потрібної інформації), що виконана згідно з вимогами до рівня «умінь» або містить незначні неточності, хід вирішення задачі правильний, але містить незначні неточності, здебільшого у розрахунку – 28-20 балів;

– «задовільно» – неповна відповідь на теоретичне питання (не менше 60 % потрібної інформації), виконана згідно з вимогами до «стереотипного» рівня та містить деякі помилки, у вирішенні задач простежуються значні помилки – 21-10 балів;

– «незадовільно» – незадовільна відповідь та неправильно вирішена задача – 0 балів.

Завдання в рамках практичного заняття оцінюються в 4 бали за такими критеріями:

– «відмінно» – повністю виконана робота (не менше 90 % потрібної інформації), надані відповідні обґрунтування та особистий погляд – 4 балів;

– «добре» – в роботі містяться певні неточності (не менше 75 % потрібної інформації), надані обґрунтування недостатньо повні – 3-2 бали;

– «задовільно» – в роботі містяться суттєві неточності (не менше 60 % потрібної інформації), робота виконана згідно з вимогами до «стереотипного» рівня та містить значні помилки – 1 бал;

– «незадовільно» – задача вирішена не правильно – 0 балів.

Календарний контроль: проводиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу. Умовою позитивного першого та другого календарного контролів є отримання не менше 50 % максимально можливого на момент відповідного календарного контролю рейтингу.

Семестровий контроль: залік. Необхідною умовою допуску до залікового оцінювання є написання модульної контрольної роботи і стартовий рейтинг не менше 30 балів.

$RC(max) = 68 + 32 = 100$ балів

$RC(min) = 30$ балів.

Залікове оцінювання. Студенти, які виконали всі умови допуску до заліку та мають рейтингову оцінку 60 і більше балів, отримують відповідну до набраного рейтингу оцінку без додаткових випробувань. Сума рейтингових балів, отриманих студентом протягом семестру, переводиться до підсумкової оцінки згідно з таблицею.

Якщо сума балів менша за 60, але виконані і зараховані практичні, МКР, студент виконує залікову роботу. Залікова робота оцінюється у 100 балів. Контрольне завдання цієї роботи складається з двох теоретичних запитань з переліку, що наданий у додатку до силабусу та задачі. Кожне запитання оцінюються в 50 балів за такими критеріями:

– «відмінно» – повна відповідь (не менше 95 % потрібної інформації), надані відповідні обґрунтування та особистий погляд – 47-50 балів;

– «добре» – достатньо повна відповідь (не менше 75 % потрібної інформації), що виконана згідно з вимогами до рівня «умінь» або містить незначні неточності – 37-46 балів;

– «задовільно» – неповна відповідь (не менше 60 % потрібної інформації), виконана згідно з вимогами до «стереотипного» рівня та містить деякі помилки – 30-36 балів;

– «незадовільно» – незадовільна відповідь – >30 балів.

Заохочувальні бали встановлюються за:

Виконання індивідуального семестрового завдання	в залежності від складності, але не більше 10 балів
Доповідь на відповідних студентських конференціях з тематики дисципліни та участь у конкурсах робіт	до 5 балів

Підготовка реферативної роботи за темою пропущеного студентом заняття, або за темою, запропонованою викладачем (обсяг до 10 аркушів)	в залежності від розкриття студентом обраної теми, обґрунтованості висновків, але не більше 3 балів
--	---

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою

<i>Рейтингові бали, RD</i>	<i>Оцінка за університетською шкалою</i>
$95 \leq RD \leq 100$	Відмінно
$85 \leq RD \leq 94$	Дуже добре
$75 \leq RD \leq 84$	Добре
$65 \leq RD \leq 74$	Задовільно
$60 \leq RD \leq 64$	Достатньо
$RD < 60$	Незадовільно
Невиконання умов допуску до семестрового контролю	Не допущено

2. Додаткова інформація з дисципліни (освітнього компонента)

В якості семестрового контролю, згідно навчального плану, студенти складають іспит. Перелік питань, які виносяться на семестровий контроль, наведено у додатку до силабусу.

Здобувач вищої освіти має можливість пройти онлайн курс(и) за однією або декількома темами, передбаченими робочою програмою навчальної дисципліни. Онлайн курс здобувач може обрати самостійно або за рекомендацією викладача. 1 год прослуханого курсу оцінюється у 0,83 бали. Максимальна кількість годин, яка може бути зарахована за результатами неформальної освіти, становить 12 год, відповідно максимальна кількість балів за такі результати становить – 10 балів.

Робочу програму навчальної дисципліни (силабус):

Складено: доцент кафедри автоматизації електротехнічних та мехатронних комплексів, к.т.н., Кулаковський Леонід Ярославович

Ухвалено кафедрою автоматизації електротехнічних та мехатронних комплексів. Протокол №27 від 23.06.26 р.

Погоджено: Методичною комісією НН ІЕЕ (протокол №43 від 29.06.26 р.)

Додаток до силабусу освітнього компонента курсу**«Основи Data Science»****Перелік завдань, що виносяться на семестровий контроль**

1. Описати роль Data Science у сучасних електротехнічних та мехатронних системах.
2. Сформулювати основні етапи процесу Data Science.
3. Навести приклади застосування аналізу даних для оптимізації роботи електроенергетичних комплексів.
4. Описати призначення Jupyter Notebook для інженерних розрахунків.
5. Описати типи даних у Python.
6. Навести приклади використання структур даних (списки, словники) для параметрів електричних схем.
7. Сформулювати принцип роботи циклів for та while для обробки масивів даних.
8. Описати призначення функцій у Python та навести приклад функції для розрахунку потужності.
9. Сформулювати поняття лямбда-виразів та навести приклад їх використання для обробки даних.
10. Описати механізм обробки винятків та його значення для надійності програм.
11. Навести приклади використання операторів break, continue у циклах.
12. Описати переваги використання NumPy для роботи з великими масивами даних.
13. Навести приклади створення масивів та матриць для моделювання електричних параметрів.
14. Сформулювати поняття векторизації та пояснити, чому вона важлива для швидких обчислень.
15. Описати методи генерації випадкових чисел та їх застосування у моделюванні шуму сигналів.
16. Навести приклади операцій над матрицями (додавання, множення, транспонування).
17. Описати принцип транслявання масивів (broadcasting) та його значення.
18. Описати структуру об'єктів Series та DataFrame та їх роль у роботі з табличними даними.
19. Навести приклади очищення даних (видалення пропусків, дублікатів) у задачах моніторингу обладнання.
20. Сформулювати принципи групування та агрегації даних.
21. Описати методи роботи з індексами та ієрархічною індексацією.
22. Навести приклади імпорту даних з CSV та Excel у DataFrame.
23. Сформулювати можливості мови запитів (query()) для вибірки даних.
24. Описати основні компоненти графіка в Matplotlib та їх призначення.
25. Навести приклади побудови графіків для аналізу трендів напруги та струму.
26. Сформулювати переваги інтерактивної візуалізації з Plotly та Seaborn для аналітичних звітів.
27. Описати методи побудови багатопанельних графіків (subplot, subplots).
28. Навести приклади використання heatmap та pairplot для аналізу кореляцій.
29. Описати можливості SciPy для статистичного аналізу даних.
30. Навести приклади застосування t-тесту або χ^2 -тесту для перевірки гіпотез у технічних вимірюваннях.
31. Сформулювати принцип роботи інтерполяції та її значення для апроксимації характеристик обладнання.
32. Описати методи фільтрації сигналів та їх застосування у системах контролю.
33. Навести приклади оптимізації функцій для підбору параметрів моделей електротехнічних процесів.
34. Сформулювати роль чисельного інтегрування та розв'язання диференціальних рівнянь у моделюванні динаміки систем.
35. Описати застосування SciPy у задачах машинного навчання (оптимізація функції втрат).