



# Інтелектуальний аналіз даних

## Робоча програма навчальної дисципліни (Силабус)

### Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	12 Інформаційні технології
Спеціальність	124 «Системний аналіз», 122 «Комп'ютерні науки»
Освітня програма	Системний аналіз і управління, Системи і методи штучного інтелекту
Статус дисципліни	Вибіркова
Форма навчання	очна(денна)
Рік підготовки, семестр	3 курс, осінній семестр
Обсяг дисципліни	4 кредити/ 120 годин (36 годин лекції, 18 годин практичні, 66 годин СРС)
Семестровий контроль/ контрольні заходи	Контрольні заходи: – поточний контроль, – календарний контроль – моніторинг поточного стану виконання вимог силабусу, – семестровий контроль: залік.
Розклад занять	<a href="https://schedule.kpi.ua/">https://schedule.kpi.ua/</a>
Мова викладання	Українська, англійська (для отримання сертифікатів від інших університетів)
Інформація про керівника курсу / викладачів	Лектор і автор курсу: професор кафедри ММСА, д.т.н., доцент Недашківська Надія Іванівна, nedashkovskaya.nadezhda@iit.kpi.ua n.nedashkivska@gmail.com Практичні заняття: професор кафедри ММСА, д.т.н., доцент Недашківська Надія Іванівна, nedashkovskaya.nadezhda@iit.kpi.ua n.nedashkivska@gmail.com
Розміщення курсу	Платформа дистанційного навчання «Сікорський», Googleclassroom, код курсу <b>y2d7l6m</b>

### Програма навчальної дисципліни

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

Метою навчальної дисципліни є формування у студентів здатностей:

- аналізу і обробки даних у різних форматах з метою підтримки прийняття рішень,
- пошуку шаблонів у великих і надвеликих базах даних,
- побудови прогнозів з використанням сучасних методів і алгоритмів інтелектуального аналізу даних, машинного навчання та інформаційних технологій;
- застосування сучасних методів інтелектуального аналізу даних та машинного навчання;
- використання програмного забезпечення для інтелектуального аналізу даних та машинного навчання в практичній роботі.

Після засвоєння навчальної дисципліни студенти мають продемонструвати такі результати навчання:

**компетентності:**

*ЗК 01 – здатність застосовувати знання в практичних ситуаціях,*

*ЗК 02 – здатність планувати і управляти часом,*

*ЗК 03 – здатність абстрактно мислити, застосовувати методи аналізу і синтезу,*

*ЗК 04 – здатність знати та розуміти предметну область і професійну діяльність,*

*ЗК 05 – здатність спілкуватися державною мовою усно і письмово,*

*ЗК 07 – здатність до пошуку, оброблення та аналізу інформації з різних джерел,*

*ЗК 09 – здатність до адаптації та дії в новій ситуації,*

*ЗК 10 – здатність системно аналізувати свою професійну і соціальну діяльність, оцінювати накопичений досвід спільно з роботодавцями та академічною спільнотою,*

*ЗК 11 – здатність генерувати нові ідеї (креативність),*

*ЗК 12 – здатність працювати в команді та автономно виконувати командні рішення,*

*ЗК 14 – здатність забезпечувати та оцінювати якість виконуваних робіт,*

*ЗК 17 – здатність ухвалювати рішення та діяти, дотримуючись принципу неприпустимості корупції та будь-яких інших проявів недоброчесності,*

*ФК 01 – здатність використовувати системний аналіз в якості сучасної міждисциплінарної методології, заснованої на прикладах математичних методів та сучасних інформаційних технологіях, і орієнтована на вирішення задач аналізу і синтезу технічних, економічних, соціальних, екологічних та інших складних систем,*

*ФК 03 – здатність будувати математично коректні моделі статичних та динамічних процесів і систем із зосередженими та розподіленими параметрами із врахуванням невизначеності зовнішніх та внутрішніх факторів,*

*ФК 06 – здатність до комп'ютерної реалізації математичних моделей реальних систем і процесів; проектувати, застосовувати і супроводжувати програмні засоби моделювання, прийняття рішень, обробки інформації, інтелектуального аналізу даних,*

*ФК 07 – здатність використовувати сучасні інформаційні технології для комп'ютерної реалізації математичних моделей та прогнозування поведінки конкретних систем, а саме: об'єктно-орієнтований підхід при проектуванні складних систем різної природи, прикладні математичні пакети, застосування баз даних і знань,*

*ФК 10 – здатність розробляти експериментальні та спостережувальні дослідження і аналізувати дані, отримані в них,*

*застосовувати методи і засоби роботи з даними і знаннями, методи математичного моделювання, технології системного аналізу,*

*проектувати, реалізовувати, тестувати, впроваджувати, супроводжувати, експлуатувати програмні засоби роботи з даними і знаннями в комп'ютерних системах і мережах,*

*розуміти і застосовувати на практиці методи статичного моделювання і прогнозування, оцінювати вихідні дані,*

**програмні результати навчання:**

*ПРН 09 – вміти створювати ефективні алгоритми для обчислювальних задач системного аналізу та систем підтримки прийняття рішень,*

*ПРН 12 – застосовувати методи і засоби роботи з даними і знаннями, методи математичного, логіко-семантичного, об'єктного та імітаційного моделювання, технології системного і статистичного аналізу,*

ПРН 13 – проектувати, реалізовувати, тестувати, впроваджувати, супроводжувати, експлуатувати програмні засоби роботи з даними і знаннями в комп'ютерних системах і мережах,

ПРН 14 – розуміти і застосовувати на практиці методи статистичного моделювання і прогнозування, оцінювати та аналізувати вихідні дані,

ПРН 17 – зберігати та примножувати досягнення і цінності суспільства на основі розуміння місця предметної області у загальній системі знань

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

При вивченні дисципліни використовуються знання дисциплін «Математичний аналіз», «Теорія ймовірностей», «Дискретна математика (розділ «Теорія графів»)», «Алгебра і геометрія», «Програмування та алгоритмічні мови», «Алгоритми та структури даних», «Математична логіка і теорія алгоритмів», «Об'єктно-орієнтоване програмування», «Практичний курс іноземної мови», «Математична статистика».

Знання, набуті при вивченні цієї дисципліни, використовуються при опануванні дисциплін «Інтелектуальні системи підтримки прийняття рішень», «Методи і технології інтелектуального аналізу даних», «Інтелектуальний аналіз великих сховищ даних», в дипломному проектуванні, у практичній самостійній роботі випускника в галузі аналізу даних під час аналізу великих і надвеликих баз даних, при побудові прогнозів на основі статистичних даних, при розробці корпоративних інформаційно-аналітичних систем в державних і приватних управлінських структурах.

## **3. Зміст навчальної дисципліни**

### **Розділ 1. Основні поняття інтелектуального аналізу даних (ІАД) та машинного навчання (МН)**

Тема 1. Загальні відомості про ІАД. Досвід в задачах МН: задачі навчання з вчителем – *supervised learning* і без вчителя – *unsupervised learning*. Класифікація, регресія, машинний переклад, пошук асоціативних правил, структурний вивід, синтез і вибірка, оцінка функції ймовірності або функції щільності ймовірності, кластеризація, сегментація. Огляд методів ІАД та машинного навчання.

Тема 2. Оцінювання якості алгоритмів МН. Перехресна перевірка (*cross-validation, CV*). Поняття перенавчання (*overfitting*) моделі МН та регуляризації моделі. Компроміс між систематичною помилкою і дисперсією моделі. Порівняння алгоритмів. Вибір гіперпараметрів моделі методами решітчастого *Grid Search CV* та рандомізованого пошуку *Random Search CV*.

Тема 3. Байєсівський підхід та підхід максимальної правдоподібності до оцінювання якості моделі.

### **Розділ 2. Методи та алгоритми класифікації**

Тема 1. Дерева рішень *Decision Trees*. Алгоритм розбиття, його властивості. Критерії вибору змінної розбиття: ентропійний, Джині. Алгоритми *ID3*, *C4.5* вибору змінної розбиття, їх властивості. Міри ефективності дерев рішень. Проблема зупинки побудови дерева.

Тема 2. Дерева рішень в *scikit-learn python*. Реалізація алгоритму розбиття *CART* для класифікації та регресії. Регуляризація дерев рішень в *scikit-learn python*. Приклад практичної задачі багатокласової класифікації ірисів Фішера. Переваги і недоліки дерев рішень. Алгоритм покриття. *1-R* алгоритм класифікації.

Тема 3. Оцінювання ефективності алгоритмів класифікації. K-Fold CV та його модифікації. Матриця неточностей (confusion matrix), метрики accuracy, precision, recall, specificity, F1-score для вибору моделі. Криві ROC-curve, PR-curve. Вибір моделі на прикладі задачі класифікації ірисів Фішера. Проблема незбалансованих класів.

Тема 4. Байєсівський підхід до класифікації. Оптимальний байєсівський класифікатор. Оцінювання апіорних імовірностей та функцій правдоподібності за вибіркою. Наївний байєсівський класифікатор. Реалізація алгоритму Naïve Bayes в scikit-learn python. Приклад практичної задачі класифікації алгоритмом Naïve Bayes в scikit-learn python.

Тема 5. Задача розділу суміші. Алгоритм Expectation-Maximization та його модифікації. Реалізація алгоритму GaussianMixture в scikit-learn python. Приклад практичної задачі класифікації алгоритмом GaussianMixture.

Тема 6. Побудова математичних функцій класифікації. Метод опорних векторів (support vector machine, SVM): лінійний та нелінійний випадки. Реалізація методу в scikit-learn python. Приклад розв'язання задачі класифікації алгоритмами SVC, NuSVC та LinearSVC scikit-learn python. Приклад розв'язання задачі регресії алгоритмами SVR, NuSVR та LinearSVR scikit-learn python. Обґрунтування методу опорних векторів. Вибір параметру розміття смуги. Функція ядра, властивості, типи ядер. Переваги і недоліки методу опорних векторів.

Тема 7. Основи теорії штучних нейронних мереж. Класичний і сучасні перцептрони (multiple layer perceptron, MLP). Функції активації: сигмоїдна, гіперболічний тангенс, ReLU, LeakyReLU та їх модифікації. Реалізація побудови і навчання багатoshарового перцептрону в scikit-learn python. Приклад розв'язання задачі класифікації набору даних MNIST алгоритмом MLPClassifier scikit-learn python. Приклад розв'язання задачі регресії – прогнозування попиту на велосипеди на основі сезонного циклічного часового ряду fetch\_openml – алгоритмом MLPRegressor scikit-learn python. Метод стохастичного градієнтного спуску. Пакетний, стохастичний (SGD) та міні-пакетний (mini-batch) варіанти реалізації градієнтного спуску (gradient descent, GD). Проблема вибору гіперпараметру швидкості навчання learning rate.

### **Розділ 3. Методи та алгоритми кластеризації**

Тема 1. Функції відстані. Ієрархічна кластеризація: агломеративний алгоритм найближчого сусіда, дівізімний алгоритм. Алгоритм AgglomerativeClustering scikit-learn python. Методи розрахунку відстані між кластерами. Приклад кластеризації наборів даних різної форми алгоритмом AgglomerativeClustering scikit-learn python. Поняття дендрограми та її побудова засобами scipy.cluster.hierarchy.

Тема 2. Алгоритми k-середніх, нечітких k-середніх та g-середніх. Реалізація в scikit-learn python: алгоритми KMeans, MiniBatchKMeans. Приклади кластеризації наборів даних різної форми, використовуючи KMeans scikit-learn python, вибір значень гіперпараметрів. Емпірична оцінка впливу ініціалізації в методі k-середніх. Порівняння алгоритмів KMeans та MiniBatchKMeans на наборах даних make\_blobs.

Тема 3. Аналіз результатів кластеризації. Метрики якості homogeneity, completeness, v-measure, adjusted rand score, adjusted mutual info score, коефіцієнт силуету. Вибір кількості кластерів за допомогою аналізу силуетів у кластеризації KMeans. Порівняння результатів, оцінювання часу виконання та якості результатів на прикладі практичної задачі кластеризації рукописних цифр з набору sklearn.datasets.load\_digits.

Тема 4. Методи кластеризації на основі теорії графів. Алгоритми Прима, Крускала і Борувки побудови мінімального покриваючого дерева. Алгоритм Форел та його модифікації.

Тема 5. Кластеризація на основі мережі Кохонена. Конкурендне навчання. Метод самоорганізуючих карт Кохонена. Інтерпретація карт Кохонена.

Тема 6. Щільнісні алгоритми кластеризації Mean Shift, DBSCAN та OPTICS. Реалізація в scikit-learn python та вибір гіперпараметрів. Аналіз результатів кластеризації наборів даних різної форми.

Алгоритми Affinity propagation, SpectralClustering та Birch.

#### **Розділ 4. Ансамблі моделей ІАД**

Тема 1. Види ансамблів. Беггінг. Композиції дерев рішень та випадковий ліс Random Forest. Реалізація в scikit-learn python: алгоритми BaggingClassifier, BaggingRegressor, RandomForestClassifier, RandomForestRegressor, ExtraTreesClassifier, ExtraTreesRegressor. Методи випадкових ділянок (random patches method) і випадкових підпросторів (random subspaces method). Переваги і недоліки випадкового лісу. Класифікатор з жорстким і м'яким голосуванням. Реалізація в scikit-learn python: VotingClassifier, VotingRegressor.

Тема 2. Бустинг. Методи AdaBoost та градієнтний бустинг. Реалізація в scikit-learn python: алгоритми AdaBoostClassifier, AdaBoostRegressor, GradientBoostingClassifier, GradientBoostingRegressor. Зміст параметрів. Переваги і недоліки бустингу. Стекінг. Реалізація в scikit-learn python: StackingClassifier, StackingRegressor.

Напрямки розвитку та перспективи подальших досліджень в області ІАД та машинного навчання. Невирішені проблеми.

#### **4. Навчальні матеріали та ресурси**

##### **Базова**

1. Н.І. Недашківська. Дистанційний курс «Інтелектуальний аналіз даних» для здобувачів першого (бакалаврського) рівня вищої освіти за освітньою програмою «Системний аналіз і управління» розміщений на платформі дистанційного навчання «Сікорський», код курсу 371zksi (Сертифікат затверджено Методичною радою КПІ ім. Ігоря Сікорського, протокол № 4 від 01.02.2024 р.)
2. Aurelien Geron. Hands-On ML with Scikit-Learn, Keras & TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 2020. <https://github.com/Akramz/Hands-on-Machine-Learning-with-Scikit-Learn-Keras-and-TensorFlow?tab=readme-ov-file>
3. Н.І. Недашківська. Інтелектуальний аналіз даних : Практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 124 «Системний аналіз», освітніх програм «Системний аналіз і управління», «Системний аналіз фінансового ринку»/ Н. І. Недашківська; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 6 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2021. – 105 с. <https://ela.kpi.ua/handle/123456789/53763>

##### **Додаткова література**

4. Jake VanderPlas. Python Data Science Handbook. Essential Tools for Working with Data. O'Reilly Media Inc., 2017. 576 p. <https://jakevdp.github.io/PythonDataScienceHandbook/>
5. Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep Learning. The MIT Press Cambridge, Massachusetts London, England, 2017. <https://www.deeplearningbook.org/>
6. Sebastian Raschka, Vahid Mirjalili. Python Machine Learning. Third Edition. Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2. Packt Publishing, 2019. <https://github.com/rasbt/python-machine-learning-book-3rd-edition>
7. Н.І. Недашківська. Методи і моделі інтелектуального аналізу даних: Практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки», освітньої програми «Системи і методи штучного інтелекту» / Н. І. Недашківська; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,8 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2019. – 70 с. <https://ela.kpi.ua/handle/123456789/53764>

8. Wes McKinney. *Python for Data Analysis*. O'Reilly Media Inc., 2013. 482 p. (за запитом викладачу)
9. Henrik Brink Joseph W. Richards Mark Fetherolf. *Real-World Machine Learning*. Manning Publications. 2016. 336 p. (за запитом викладачу)
10. Andreas C. Mueller and Sarah Guido. *An Introduction to Machine Learning with Python*. O'Reilly Media Inc., 2017. 392 p. (за запитом викладачу)
11. Davy Cielen, Arno Meysman, Mohamed Ali. *Introducing Data Science: Big Data, Machine Learning, and more, using Python tools*. Manning Publications, 2016. 320 p. (за запитом викладачу)
12. Scikit-Learn Documentation. Режим доступу: <https://scikit-learn.org/>, 2025.

Використовується сучасне комп'ютерне та мультимедійне обладнання, платформа дистанційного навчання «Сікорський».

Для виконання практичних робіт використовується open-source програмне забезпечення Python (<https://www.python.org/>), Scikit-Learn 1.2.1 – open source, commercially usable – BSD license (<https://scikit-learn.org/>), TensorFlow v.2.11.0 – Apache-2.0 license (<https://www.tensorflow.org/>), Keras – Apache-2.0 license (<https://keras.io>)

## Навчальний контент

### 5. Методика опанування навчальної дисципліни(освітнього компонента)

#### Структура кредитного модуля

Назви розділів і тем	Кількість годин				
	Всього	у тому числі			
		Лекції	Семінарські	Практичні (комп'ютерний практикум)	СРС
1	2	3	4	5	6
<b>Розділ 1. Основні поняття інтелектуального аналізу даних (ІАД) та машинного навчання (МН)</b>					
Тема 1. Загальні відомості про ІАД. Досвід в задачах МН: задачі навчання з вчителем – supervised learning і без вчителя – unsupervised learning. Класифікація, регресія, машинний переклад, пошук асоціативних правил, структурний вивід, синтез і вибірка, оцінка функції ймовірності або функції щільності ймовірності, кластеризація, сегментація. Огляд методів ІАД та машинного навчання.	4	2			2
Тема 2. Оцінювання якості алгоритмів МН. Перехресна перевірка (cross-validation, CV). Поняття перенавчання (overfitting) моделі МН та регуляризації моделі. Компроміс між систематичною помилкою і дисперсією моделі. Криві навчання і перевірки. U-крива. Порівняння алгоритмів. Вибір гіперпараметрів моделі методами решітчастого Grid Search CV та рандомізованого пошуку Random Search CV.	7	3		2	2
Тема 3. Байєсівський підхід та підхід максимальної правдоподібності до оцінювання якості моделі.	5	1			4
<b>Разом за розділом 1</b>	<b>16</b>	<b>6</b>		<b>2</b>	<b>8</b>
<b>Розділ 2. Методи та алгоритми класифікації</b>					

1	2	3	4	5	6
Тема 1. Древа рішень Decision Trees. Алгоритм розбиття, його властивості. Критерії вибору змінної розбиття: ентропійний, Джині. Алгоритми ID3, C4.5 вибору змінної розбиття, їх властивості. Міри ефективності дерев рішень. Проблема зупинки побудови дерева.	5	2		1	2
Тема 2. Древа рішень в scikit-learn python. Реалізація алгоритму розбиття CART для класифікації та регресії. Регуляризація дерев рішень в scikit-learn python. Приклад практичної задачі багатокласової класифікації ірисів Фішера. Переваги і недоліки дерев рішень. Алгоритм покриття. 1-R алгоритм класифікації.	7	2		1	4
Тема 3. Оцінювання ефективності алгоритмів класифікації. K-Fold CV та його модифікації. Матриця неточностей (confusion matrix), метрики accuracy, precision, recall, specificity, F1-score для вибору моделі. Криві ROC-curve, PR-curve. Вибір моделі на прикладі задачі класифікації ірисів Фішера. Проблема незбалансованих класів.	6	2		2	2
Тема 4. Байесівський підхід до класифікації. Оптимальний байесівський класифікатор. Оцінювання апіорних імовірностей та функцій правдоподібності за вибіркою. Наївний байесівський класифікатор. Реалізація алгоритму Naïve Bayes в scikit-learn python. Приклад практичної задачі класифікації алгоритмом Naïve Bayes в scikit-learn python.	9	2		1	6
Тема 5. Задача розділу суміші. Алгоритм Expectation-Maximization та його модифікації. Реалізація алгоритму GaussianMixture в scikit-learn python. Приклад практичної задачі класифікації алгоритмом GaussianMixture.	7	2		1	4
Тема 6. Побудова математичних функцій класифікації. Метод опорних векторів (support vector machine, SVM): лінійний та нелінійний випадки. Реалізація методу в scikit-learn python. Приклад розв'язання задачі класифікації алгоритмами SVC, NuSVC та LinearSVC scikit-learn python. Приклад розв'язання задачі регресії алгоритмами SVR, NuSVR та LinearSVR scikit-learn python. Обґрунтування методу опорних векторів. Вибір параметру розмиття смуги. Функція ядра, властивості, типи ядер. Переваги і недоліки методу опорних векторів.	7	2		1	4
Тема 7. Основи теорії штучних нейронних мереж. Класичний і сучасні перцептрони (multiple layer perceptron, MLP). Функції активації: сигмоїдна, гіперболічний тангенс, ReLU, LeakyReLU та їх модифікації. Реалізація побудови і навчання багатoshарового перцептрону в scikit-learn python. Приклад розв'язання задачі класифікації набору даних MNIST алгоритмом MLPClassifier scikit-learn python. Приклад розв'язання задачі регресії – прогнозування попиту на велосипеди на основі сезонного циклічного часового ряду fetch_openml –	7	2		1	4

1	2	3	4	5	6
алгоритмом MLPRegressor scikit-learn python. Метод стохастичного градієнтного спуску. Пакетний, стохастичний (SGD) та міні-пакетний (mini-batch) варіанти реалізації градієнтного спуску (gradient descent, GD). Проблема вибору гіперпараметру швидкості навчання learning rate.					
Перша частина модульної контрольної роботи	4				4
<b>Разом за розділом 2</b>	<b>52</b>	<b>14</b>		<b>8</b>	<b>30</b>
<b>Розділ 3. Методи та алгоритми кластеризації</b>					
Тема 1. Функції відстані. Ієрархічна кластеризація: агломеративний алгоритм найближчого сусіда, дівізимний алгоритм. Алгоритм AgglomerativeClustering scikit-learn python. Методи розрахунку відстані між кластерами. Приклад кластеризації наборів даних різної форми алгоритмом AgglomerativeClustering scikit-learn python. Поняття дендрограми та її побудова засобами scipy.cluster.hierarchy.	5	2		1	2
Тема 2. Алгоритми k-середніх, нечітких k-середніх та g-середніх. Реалізація в scikit-learn python: алгоритми KMeans, MiniBatchKMeans. Приклади кластеризації наборів даних різної форми, використовуючи KMeans scikit-learn python, вибір значень гіперпараметрів. Емпірична оцінка впливу ініціалізації в методі k-середніх. Порівняння алгоритмів KMeans та MiniBatchKMeans на наборах даних make_blobs.	6	2		1	3
Тема 3. Аналіз результатів кластеризації. Метрики якості homogeneity, completeness, v-measure, adjusted rand score, adjusted mutual info score, коефіцієнт силуету. Вибір кількості кластерів за допомогою аналізу силуетів у кластеризації KMeans. Порівняння результатів, оцінювання часу виконання та якості результатів на прикладі практичної задачі кластеризації рукописних цифр з набору sklearn.datasets.load_digits.	6	2		1	3
Тема 4. Методи кластеризації на основі теорії графів. Алгоритми Прима, Крускала і Борувки побудови мінімального покриваючого дерева. Алгоритм Форел та його модифікації.	4	2			2
Тема 5. Кластеризація на основі мережі Кохонена. Конкурентне навчання. Метод самоорганізуючих карт Кохонена. Інтерпретація карт Кохонена.	4	2			2
Тема 6. Щільнісні алгоритми кластеризації Mean Shift, DBSCAN та OPTICS. Реалізація в scikit-learn python та вибір гіперпараметрів. Аналіз результатів кластеризації наборів даних різної форми.  Алгоритми Affinity propagation, SpectralClustering та Birch.	5	2		1	2
<b>Разом за розділом 3</b>	<b>30</b>	<b>12</b>		<b>4</b>	<b>14</b>
<b>Розділ 4. Ансамблі моделей ІАД</b>					

1	2	3	4	5	6
<p>Тема 1. Види ансамблів. Бегінг. Композиції дерев рішень та випадковий ліс Random Forest. Реалізація в scikit-learn python: алгоритми BaggingClassifier, BaggingRegressor, RandomForestClassifier, RandomForestRegressor, ExtraTreesClassifier, ExtraTreesRegressor. Методи випадкових ділянок (random patches method) і випадкових підпросторів (random subspaces method). Переваги і недоліки випадкового лісу.</p> <p>Класифікатор з жорстким і м'яким голосуванням. Реалізація в scikit-learn python: VotingClassifier, VotingRegressor.</p>	10	2		2	6
<p>Тема 2. Бустинг. Методи AdaBoost та градієнтний бустинг. Реалізація в scikit-learn python: алгоритми AdaBoostClassifier, AdaBoostRegressor, GradientBoostingClassifier, GradientBoostingRegressor. Зміст параметрів. Переваги і недоліки бустингу.</p> <p>Стекінг. Реалізація в scikit-learn python: StackingClassifier, StackingRegressor. Практична задача: прогнозування і оцінювання важливості ознак (features) за допомогою RandomForestClassifier. Порівняння різних класифікаторів на основі бегінгу, бустингу і стекінгу, їх метрик якості, кривих ROC і DET для трьох різних навчальних наборів даних: make_moons, make_circles, make_classification.</p> <p>Напрямки розвитку та перспективи подальших досліджень в області ІАД та машинного навчання. Невирішені проблеми.</p>	12	2		2	8
Друга частина модульної контрольної роботи					
<b>Разом за розділом 4</b>	<b>22</b>	<b>4</b>		<b>4</b>	<b>14</b>
<b>Всього годин</b>	<b>120</b>	<b>36</b>		<b>18</b>	<b>66</b>

### Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на літературу та завдання на СРС)
1	<p>Тема 1.1. Загальні відомості про ІАД. Досвід в задачах МН: задачі навчання з вчителем – supervised learning і без вчителя – unsupervised learning. Класифікація, регресія, машинний переклад, пошук асоціативних правил, структурний вивід, синтез і вибірка, оцінка функції ймовірності або функції щільності ймовірності, кластеризація, сегментація. Огляд методів ІАД та машинного навчання. [1, 2, 4, 6]</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Розв'язання практичних задач в сфері інтернет-технологій. Методи і технології Web Mining [4, 8]</p>
2	<p>Тема 1.2. Оцінювання якості алгоритмів МН. Перехресна перевірка (cross-validation, CV). Поняття перенавчання (overfitting) моделі МН та регуляризації моделі. Компроміс між систематичною помилкою і дисперсією моделі. Криві навчання і перевірки. U-крива. Порівняння алгоритмів. Вибір гіперпараметрів моделі методами решітчастого Grid Search CV та рандомізованого пошуку Random Search CV. [1 – 11]</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Етапи процесу виявлення знань. Управління знаннями (knowledge</p>

	<i>management). Метод, орієнтований на продукти. Метод, орієнтований на процеси. [8, 9]</i>
3	<p>Тема 1.3. Теорема Байеса. Максимальна апостеріорна гіпотеза. Метод максимальної правдоподібності. [1, 2, 6]</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Задача лінійної регресії. Міра якості в задачі лінійної регресії. Функція помилки в задачі регресії та її обґрунтування. Регуляризація зі зниженням ваги та її обґрунтування за допомогою байєсівського підходу. Гребнева регресія. [1 – 4] Лассо регресія. Еластична регресія [4, 5].</p>
4	<p>Тема 2.1. Дерева рішень Decision Trees. Алгоритм розбиття, його властивості. Критерії вибору змінної розбиття: ентропійний, Джині. Алгоритми ID3, C4.5 вибору змінної розбиття, їх властивості. Міри ефективності дерев рішень. Проблема зупинки побудови дерева. [1 – 5]</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Вибір параметрів методу Decision Trees. Представлення результатів [5, 11]</p>
5	<p>Тема 2.2. Дерева рішень в scikit-learn python [5]. Реалізація алгоритму розбиття CART для класифікації та регресії. Регуляризація дерев рішень в scikit-learn python. Приклад практичної задачі багатокласової класифікації ірисів Фішера. Переваги і недоліки дерев рішень [1, 4]</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Алгоритм покриття. 1-R алгоритм класифікації. [1, 2]</p>
6	<p>Тема 2.3. Оцінювання ефективності алгоритмів класифікації. K-Fold CV та його модифікації. Матриця неточностей (confusion matrix), метрики accuracy, precision, recall, specificity, F1-score для вибору моделі. [1 – 5] Криві ROC-curve, PR-curve. Вибір моделі на прикладі задачі класифікації ірисів Фішера. Проблема незбалансованих класів. [1, 2]</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Крива DET-curve [5]. Приклади розв'язання практичних задач [5, 11].</p>
7	<p>Тема 2.4. Байєсівський підхід до класифікації. Оптимальний байєсівський класифікатор. Оцінювання апіорних імовірностей та функцій правдоподібності за вибіркою. [1, 2] Наївний байєсівський класифікатор. Реалізація алгоритму Naïve Bayes в scikit-learn python [5]. Приклад практичної задачі класифікації алгоритмом Naïve Bayes в scikit-learn python.</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p>
8	<p>Тема 2.5. Задача розділу суміші. Алгоритм Expectation-Maximization та його модифікації. [1 – 3] Реалізація алгоритму GaussianMixture в scikit-learn python [5]. Приклад практичної задачі класифікації алгоритмом GaussianMixture.</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Реалізація в scikit-learn python.</p>
9	<p>Тема 2.6. Побудова математичних функцій класифікації. Метод опорних векторів (support vector machine, SVM): лінійний та нелінійні випадки. [1, 2] Реалізація методу в scikit-learn python [3, 5]. Приклад розв'язання задачі класифікації алгоритмами SVC, NuSVC та LinearSVC scikit-learn python. Приклад розв'язання задачі регресії алгоритмами SVR, NuSVR та LinearSVR scikit-learn python [5]. Обґрунтування методу опорних векторів. [1, 2] Вибір параметру розбиття смуги. Функція ядра, властивості [4]. Переваги і недоліки методу опорних векторів.</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Вибір типу ядра в SVM [4, 5]. Налаштування параметрів методу SVM [1, 3, 4, 8]</p>
10	<p>Тема 2.7. Основи теорії штучних нейронних мереж. Класичний і сучасні перцептрони (multiple layer perceptron, MLP). Функції активації: сигмоїдна, гіперболічний тангенс, ReLU, LeakyReLU та їх модифікації. [1, 2, 6, 7] Реалізація побудови і навчання</p>

	<p>багатошарового перцептрон у <code>scikit-learn python</code> [5]. Приклад розв'язання задачі класифікації набору даних MNIST алгоритмом <code>MLPClassifier scikit-learn python</code>. Приклад розв'язання задачі регресії – прогнозування попиту на велосипеди на основі сезонного циклічного часового ряду <code>fetch_openml</code> – алгоритмом <code>MLPRegressor scikit-learn python</code>. Метод стохастичного градієнтного спуску. Пакетний, стохастичний (SGD) та міні-пакетний (mini-batch) варіанти реалізації градієнтного спуску (<code>gradient descent, GD</code>). Проблема вибору гіперпараметру швидкості навчання <code>learning rate</code> [1,2] Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Адаптивні алгоритми градієнтного спуску [6, 7]. Функція активації <code>Swish</code>.</p>
11	<p>Тема 3.1. Функції відстані. Ієрархічна кластеризація: агломеративний алгоритм найближчого сусіда, дивізимний алгоритм. [1 – 4] Алгоритм <code>AgglomerativeClustering scikit-learn python</code>. Методи розрахунку відстані між кластерами. Приклад кластеризації наборів даних різної форми алгоритмом <code>AgglomerativeClustering scikit-learn python</code> [5]. Поняття дендрограми.</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> побудова дендрограми засобами <code>scipy.cluster.hierarchy</code> [5].</p>
12	<p>Тема 3.2. Алгоритми <i>k</i>-середніх, нечітких <i>k</i>-середніх та <i>g</i>-середніх. [1 – 3, 8 – 11] Реалізація в <code>scikit-learn python</code>: алгоритми <code>KMeans</code>, <code>MiniBatchKMeans</code> [5]. Приклади кластеризації наборів даних різної форми, використовуючи <code>KMeans scikit-learn python</code>, вибір значень гіперпараметрів. Емпірична оцінка впливу ініціалізації в методі <i>k</i>-середніх. Порівняння алгоритмів <code>KMeans</code> та <code>MiniBatchKMeans</code> на наборах даних <code>make_blobs</code>.</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Порівняння алгоритмів <code>KMeans</code> та <code>MiniBatchKMeans</code> на різних наборах даних.</p>
13	<p>Тема 3.3. Аналіз результатів кластеризації. Метрики якості <code>homogeneity</code>, <code>completeness</code>, <code>v-measure</code>, <code>adjusted rand score</code>, <code>adjusted mutual info score</code>, коефіцієнт силуету. [1, 2, 5] Вибір кількості кластерів за допомогою аналізу силуетів у кластеризації <code>KMeans</code>. Порівняння результатів, оцінювання часу виконання та якості результатів на прикладі практичної задачі кластеризації рукописних цифр з набору <code>sklearn.datasets.load_digits</code>.</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p>
14	<p>Тема 3.4. Методи кластеризації на основі теорії графів. Алгоритми Прима, Крускала і Борувки побудови мінімального покриваючого дерева. [1]</p> <p>Алгоритм Форел та його модифікації. [1]</p> <p>Кластеризація на основі мережі Кохонена. Конкурендне навчання. Метод самоорганізуючих карт Кохонена. Інтерпретація карт Кохонена. [1]</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Розв'язання практичної задачі в <code>ППП Deductor</code>.</p>
15	<p>Тема 3.5. Щільнісні алгоритми кластеризації <code>Mean Shift</code>, <code>DBSCAN</code> та <code>OPTICS</code>. [1, 2, 4] Реалізація в <code>scikit-learn python</code> та вибір гіперпараметрів [5]. Аналіз результатів кластеризації наборів даних різної форми.</p> <p>Алгоритми <code>Affinity propagation</code>, <code>SpectralClustering</code> та <code>Birch</code>.</p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> Реалізація алгоритмів в <code>scikit-learn python</code>.</p>
16	<p>Тема 4.1. Види ансамблів. Беггінг. Композиції дерев рішень та випадковий ліс <code>Random Forest</code>. [1 – 4] Реалізація в <code>scikit-learn python</code>: алгоритми <code>BaggingClassifier</code>, <code>BaggingRegressor</code>, <code>RandomForestClassifier</code>, <code>RandomForestRegressor</code>, <code>ExtraTreesClassifier</code>, <code>ExtraTreesRegressor</code>. [5] Методи випадкових ділянок (<code>random patches method</code>) і випадкових підпросторів (<code>random subspaces method</code>). Переваги і недоліки випадкового лісу.</p>

	<p><i>Класифікатор з жорстким і м'яким голосуванням. Реалізація в scikit-learn python: VotingClassifier, VotingRegressor. [5]</i></p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> <i>Настройка параметрів. Інтерпретація результатів [4, 5, 8 – 11]</i></p>
17	<p><i>Тема 4.2. Бустинг. Методи AdaBoost та градієнтний бустинг. [1 – 4] Реалізація в scikit-learn python: алгоритми AdaBoostClassifier, AdaBoostRegressor, GradientBoostingClassifier, GradientBoostingRegressor. [5] Зміст параметрів. Переваги і недоліки бустингу.</i></p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p> <p><b>Завдання на СРС:</b> <i>Настройка параметрів. Інтерпретація результатів [4, 5, 8 – 11]</i></p>
18	<p><i>Тема 4.3. Стекінг. [1 – 4] Реалізація в scikit-learn python: StackingClassifier, StackingRegressor. [5]</i></p> <p><i>Практична задача: прогнозування і оцінювання важливості ознак (features) за допомогою RandomForestClassifier. Порівняння різних класифікаторів на основі бегінгу, бустингу і стекінгу, їх метрик якості, кривих ROC і DET для трьох різних навчальних наборів даних: make_moons, make_circles, make_classification.</i></p> <p><i>Методи розрахунку коефіцієнтів відносних важливостей (ваг) моделей ансамблю [1]</i></p> <p><i>Напрямки розвитку та перспективи подальших досліджень в області ІАД та машинного навчання. Невирішені проблеми.</i></p> <p>Наочність навчальних занять забезпечується використанням слайдів.</p>

### Практичні заняття

Метою практичних занять (комп'ютерного практикуму) є закріплення теоретичних положень навчальної дисципліни, отримання практичних навичок використання методів інтелектуального аналізу даних і машинного навчання з метою підтримки прийняття рішень. В результаті виконання робіт студенти повинні вміти будувати прогнози на основі статистичних даних, використовуючи вказані методи, розв'язувати практичні задачі класифікації і кластеризації з використанням сучасного програмного забезпечення Scikit-Learn Python, системно вирішувати практичні задачі аналізу і пошуку шаблонів у великих і надвеликих базах даних, використовувати сучасне програмне забезпечення Scikit-Learn Python для інтелектуального аналізу даних та машинного навчання в практичній роботі.

№ з/п	Назва роботи	Кількість ауд. годин
1	Отримання навичок роботи в середовищі Python	2
2	Побудова та оцінювання якості моделей дерев рішень та опорних векторів для класифікації та регресії засобами бібліотеки Scikit-Learn Python	4
3	Класифікація та регресія на основі багатошарового перцептрона в Scikit-Learn Python	2
4	Побудова та оцінювання якості моделей кластеризації в Scikit-Learn Python	6
5	Побудова та оцінювання ансамблів моделей класифікації та регресії засобами Scikit-Learn Python	4

Для виконання практичних робіт використовується open-source програмне забезпечення Python (<https://www.python.org/>), Scikit-Learn 1.2.1 – open source, commercially usable – BSD license (<https://scikit-learn.org/>), TensorFlow v.2.11.0 – Apache-2.0 license (<https://www.tensorflow.org/>), Keras – Apache-2.0 license (<https://keras.io>)

Для виконання практичних робіт (комп'ютерного практикуму) студентам пропонується формувати команди, що дозволяє студентам додатково тренувати власні soft-skills.

## 6. Самостійна робота студента

Самостійна робота студента включає підготовку до практичних робіт, підготовку до модульної контрольної роботи, в тому числі опрацювання тем, які не увійшли до лекцій та розв'язок задач.

Назви тем на самостійне опрацювання та кількість годин СРС наведено вище в описі структури кредитного модуля в розділі 5 «Методика опанування навчальної дисципліни (освітнього компонента)».

## Політика та контроль

### 7. Політика навчальної дисципліни (освітнього компонента)

**Правила відвідування занять.** Очікується, що студент відвідуватиме кожне аудиторне заняття, робитиме це вчасно, без запізнь. У разі особливих обставин студент, за узгодженням з викладачем, може бути відсутнім на занятті, але це не може бути систематично.

**Правила поведінки на заняттях.** На заняттях передбачається активність студентів, пов'язана з навчальним процесом, включення в інтерактивні форми та методи навчання.

Під час виконання лабораторних/практичних завдань, окрім наявного в аудиторії обладнання, студент може користуватися власним ноутбуком.

**Правила виконання і захисту практичних/ лабораторних робіт.** До захисту роботи допускаються студенти, які правильно виконали лабораторне/практичне завдання. Захист проходить у форматі індивідуального спілкування студента з викладачем по тематиці роботи та письмового виконання завдання.

**Пропущені контрольні заходи оцінювання.** Кожен студент має право відпрацювати пропущені з поважної причини (лікарняний, мобільність тощо) заняття за рахунок самостійної роботи. Детальніше за посиланням: <https://kpi.ua/files/n3277.pdf>.

**Правила призначення заохочувальних та штрафних балів.** За узгодженням з викладачем студент може зробити доповідь по темі, релевантній дисципліні, виконувати додаткові завдання. Заохоченням до подібної успішної роботи є додаткові рейтингові бали загальним обсягом до 10% від максимального рейтингового балу шкали оцінювання. Кількість та правила нарахування балів узгоджується викладачем у кожному окремому випадку.

**Політика дедлайнів та перескладань.** Студент, який з будь-яких причин не зміг вчасно виконати та захистити лабораторну/практичну роботу, може це зробити на наступних заняттях за умови доступності обладнання та часу у викладачів. Під час виконання та захисту лабораторних/практичних робіт пріоритет надається студентам, які виконують завдання згідно календарного плану. Виконання та/або захист лабораторної/практичної роботи після відведеного на неї строку призводить до зниження максимального балу, який студент може отримати за цей вид роботи. Допуск до перескладання заходів семестрового контролю та самі перескладання здійснюються згідно правил Університету у терміни, визначені Університетом.

**Процедура оскарження результатів контрольних заходів оцінювання.** Студент може підняти будь-яке питання, яке стосується процедури контрольних заходів та очікувати, що воно буде розглянуто згідно із наперед визначеними процедурами. Студенти мають право аргументовано

оскаржити результати контрольних заходів, пояснивши з яким критерієм не погоджуються відповідно до оціночного.

**Політика щодо академічної доброчесності.** Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>. У разі виявлення порушення академічної доброчесності результати роботи студента, які стосуються недоброчесності, анулюються.

**Норми етичної поведінки.** Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>.

**Інклюзивне навчання.** Засвоєння знань та умінь в ході вивчення дисципліни може бути доступним для більшості осіб з особливими освітніми потребами, окрім здобувачів з серйозними вадами зору, які не дозволяють виконувати завдання за допомогою персональних комп'ютерів, ноутбуків та/або інших технічних засобів.

**Навчання іноземною мовою.** У ході виконання завдань студентам може бути рекомендовано звернутися до англомовних джерел.

**Календарний контроль** проводиться з метою підвищення якості навчання студентів та моніторингу виконання студентом вимог силабусу.

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

**Поточний контроль:** виконання та захист 5 практичних робіт та виконання модульної контрольної роботи, участь в опитуваннях на лекціях.

**Календарний контроль:** проводиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

**Семестровий контроль:** залік.

**Умови допуску до семестрового контролю:** семестровий рейтинг не менше 40 балів.

### Рейтингова система оцінювання результатів навчання

Рейтинг студента з навчальної дисципліни складається з балів, які він отримує за результатами поточного контролю (max 100):

- 1) виконання та захист 5 практичних робіт;
- 2) виконання модульної контрольної роботи у двох частинах;
- 3) участь в опитуваннях на лекціях.

**1. Практичні роботи.** Упродовж семестру студент має виконати 5 практичних робіт (ПР). Рейтингова оцінка кожної ПР складається з 2 частин, які оцінюються окремо.

#### Критерії оцінювання

а. Якість підготовки до роботи, її виконання та оформлення звіту.

- за умови правильно оформленого звіту з точним виконанням завдання ПР – 6,5 -7 балів;
- за наявності несуттєвих неточностей в оформленні або процедурі виконання ПР – 5-6 балів;
- за наявності порушень в оформленні, неповного або неточного виконання – 3-4 бали;
- завдання не виконано, незадовільна відповідь – 0-2 бали.

б. Якість захисту матеріалу. В цій частині оцінюється ступінь володіння теоретичним і практичним матеріалом за темою роботи.

- відмінне володіння матеріалом – 6,5 – 7 балів;
- добре володіння матеріалом – 5 – 6 балів;

- задовільне володіння матеріалом – 4 бали;
- незадовільна відповідь – 0-2 бали.

	ПР – 1	ПР – 2	ПР – 3	ПР – 4	ПР – 5
Строк здачі та захисту роботи	19.09	10.10	28.10	13.11	27.11

За несвоєчасну подачу звіту з ЛР та несвоєчасний захист ПР (пізніше встановленого строку без поважної причини) можуть бути нараховані штрафні бали: мінус 0.3 бали за кожний тиждень запізнення. Штрафні бали не входять до основної шкали РСО, а їх сума не може перевищувати 10% рейтингової шкали, тобто 10 балів.

**Максимальна кількість балів за всі ПР дорівнює: 5\*14 =70 балів.**

**2. Модульна контрольна робота.** Модульна контрольна робота складається з двох частин – КР-1 і КР-2, включає теоретичні питання і задачі.

Критерії оцінювання

Оцінки за теоретичні питання визначаються за шкалою:

- «відмінно», повна відповідь (не менше 95% потрібної інформації) – 4,75 - 5 балів;
- «добре», достатньо повна відповідь (не менше 75% потрібної інформації), або повна відповідь з незначними неточностями – 4 - 4,5 бали;
- «задовільно», неповна відповідь (не менше 60% потрібної інформації) та незначні помилки – 3 - 3,5 бали;
- «незадовільно», незадовільна відповідь – 0-2 бали.

Оцінки за задачі визначаються за шкалою:

- завдання виконано правильно – 4,5 - 5 балів;
- завдання виконано з незначними помилками або неточностями – 4 бали;
- завдання виконано не повністю або з суттєвими помилками – 3 бали;
- завдання не виконано, незадовільна відповідь – 0-2 бали.

Максимальна оцінка за КР-1 складає 10 балів, КР-2 – 15 балів.

**Максимальна кількість балів за дві частини модульної КР складає 25 балів.**

	КР – 1	КР – 2
Дата проведення роботи	09.10	04.11

**3. Максимальна кількість балів за участь в опитуваннях на лекціях складає 5 балів.**

За результатами навчальної роботи за перші 6 тижнів станом на 17.10 «ідеальний студент» має набрати 43 бали, які включають дві практичні роботи та першу частину МКР. **На першому календарному контролі на 7-му тижні студент отримує «зараховано», якщо його поточний рейтинг не менше 22 балів.**

**Перший календарний контроль: 13.10 – 19.10.**

За результатами 12 тижнів навчання станом на 28.11 «ідеальний студент» має набрати 85 балів, що включають п'ять зданих і захищених практичних роботи та першу частину МКР. **На другому календарному контролі на 13-му тижні студент отримує «зараховано», якщо його поточний рейтинг не менше 43 балів.**

**Другий календарний контроль: 24.11 – 30.11.**

**Максимальна сума балів за роботу в семестрі складає 100.** Необхідною умовою допуску до заліку є отримання рейтингу 40 балів і вище. 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** («автоматом»).

Студенти, які виконали всі умови допуску до заліку та мають рейтингову оцінку менше 60 балів, а також ті студенти, хто хоче підвищити оцінку, виконують залікову контрольну роботу. При цьому оцінки за КР-1 і КР-2 скасовуються, а всі інші оцінки з дисципліни зберігаються. Оцінка за залікову контрольну роботу додається до оцінок за практикуми та опитування на лекціях, і ця сумарна оцінка є остаточною.

#### **Заліковий тиждень: 15.12 – 21.12. Залікова контрольна робота 15.12.2025.**

Завдання залікової контрольної роботи складається з двох теоретичних питань і двох практичних завдань різних розділів робочої програми.

#### Критерії оцінювання

Кожне теоретичне і практичне питання залікової контрольної роботи оцінюється у 6,25 балів відповідно до системи оцінювання:

- «відмінно», повна відповідь (не менше 95% потрібної інформації) – 5,9 - 6,25 балів;
- «добре», достатньо повна відповідь (не менше 75% потрібної інформації або незначні неточності) – 4,6 - 5,9 балів;
- «задовільно», неповна відповідь (не менше 60% потрібної інформації та деякі помилки) – 3,75 - 4,6 балів;
- «незадовільно», незадовільна відповідь – 0 - 2 бали.

**Максимальна кількість балів за залікову контрольну роботу складає 25 балів.**

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

<i>Кількість балів</i>	<i>Оцінка</i>
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Невиконання умов допуску до семестрового контролю (менше 40)	Не допущено

### **9. Додаткова інформація з дисципліни (освітнього компонента)**

*Перелік питань, які виносяться на семестровий контроль, наведено в додатку А.*

*Сертифікати проходження дистанційних чи онлайн курсів за тематикою дисципліни за попереднім узгодженням з викладачем можуть бути зараховані з додатковими 5 балами до загального рейтингу студента.*

#### **Робочу програму навчальної дисципліни (силабус):**

**Складено** професором кафедри математичних методів системного аналізу, д.т.н., доцентом Недашківською Надією Іванівною

**Ухвалено** кафедрою ММСА НН ІПСА (протокол № 3 від 08.10.2025)

**Погоджено** Методичною комісією НН ІПСА (протокол № 2 від 09.10.2025)

## Перелік питань з дисципліни

## «Інтелектуальний аналіз даних», третій курс

кафедра ММСА НН ІПСА НТУУ «КПІ ім. Ігоря Сікорського»  
викладач професор кафедри ММСА, д.т.н., доц. Недашківська Н.І.

1. Задачі машинного навчання (МН). \*
2. Задачі навчання з учителем і без учителя. \*
3. Перенавчання (overfitting) моделі МН. Ємність моделі. \*
4. Компромiс між систематичною помилкою і дисперсією.
5. Крива перевірки.
6. Теорема Байеса і максимальна апостеріорна гіпотеза.
7. Лінійна регресія. \*
8. Постановка задачі та ідея методу опорних векторів для лінійно роздільного випадку. Функціонал помилок. Означення опорного вектору. \*
9. Теоретичне обґрунтування методу опорних векторів.
10. Метод опорних векторів для лінійно нероздільного випадку. Ідея методу. Типи векторів-порушників роздільної смуги.
11. Метод опорних векторів для лінійно нероздільного випадку. Обґрунтування методу.
12. Нелінійне узагальнення методу опорних векторів. Означення ядра і приклади. Обґрунтування методу.
13. Функції класифікації за методом опорних векторів для лінійно роздільного і нероздільного випадків. Переваги і обмеження методу. Як визначається параметр  $C$ ? \*
14. Поняття дерева рішень. Структура дерева рішень. Листові і проміжні вершини дерева. Навести приклад. \*
15. Алгоритм розбиття побудови дерев рішень. Властивості алгоритму розбиття. \*
16. Означення ентропії. Ентропійний критерій вибору змінної розбиття. \*
17. Проблема перенавчання (зверхчутливості) дерев рішень. Алгоритм  $C4.5$  вибору змінної розбиття. Модифікований  $C4.5$  для випадку неперервних атрибутів.
18. Дерева рішень для класифікації в Scikit-Learn Python. Атрибути вузла дерева. Міра забрудненості Джині. \*
19. Алгоритм CART для класифікації. Алгоритм розбиття побудови дерев рішень класифікації \*
20. Переваги і обмеження алгоритму розбиття. Регуляризація дерев рішень. Міри ефективності і проблема зупинки побудови дерева рішень \*
21. Дерева рішень для регресії в Scikit-Learn Python, атрибути вузла. Алгоритм розбиття CART для регресії \*
22. Алгоритм покриття побудови дерев рішень. 1R алгоритм \*

23. Основи байесівської класифікації. Максимум апостеріорної імовірності. Штраф за помилку. Середній ризик. Оцінювання апріорних імовірностей та функцій правдоподібності
24. Наївний байесівський класифікатор (алгоритм Naive Bayes) \*
25. Оцінка ефективності класифікації: перехресна перевірка, K-fold та його модифікації, решітчатий і рандомізований пошук \*
26. Оцінка ефективності класифікації: confusion matrix, accuracy, precision, recall, PR-curve, F1 score, ROC крива \*
27. Постановка задачі розділу суміші. Базовий алгоритм EM: ідея та етапи \*
28. Обґрунтування базового алгоритму EM (виведення формул)
29. Алгоритм EM з фіксованою кількістю компонент. Недоліки алгоритму
30. Узагальнений алгоритм EM. Стохастичний алгоритм EM
31. Алгоритм EM з послідовним додаванням компонент.
32. Постановка задачі класифікації на два класи, поняття відступу, функціоналу помилок та функції втрат. Лінійний класифікатор.
33. Методи градієнтного спуску і стохастичного градієнтного спуску. \*
34. Алгоритм стохастичного градієнтного спуску: етапи, оцінка функціоналу якості.
35. Ініціалізація wag, порядок пред'явлення об'єктів в алгоритмі стохастичного градієнта. Переваги і недоліки, проблема перенавчання цього алгоритму.
36. Загальна постановка задачі кластеризації. Функції відстані в задачах кластеризації. \*
37. Агломеративний ієрархічний алгоритм найближчого сусіда. \*
38. Розділяючий ієрархічний алгоритм DIANA.
39. Алгоритм k-середніх. Переваги і недоліки алгоритму k-середніх. \*
40. Алгоритм нечітких k-середніх. Переваги і недоліки.
41. Алгоритм G-середніх.
42. Алгоритм знаходження зв'язних компонент. Переваги і недоліки.
43. Базові принципи побудови мінімального покриваючого дерева (МПД) «Жадібний» алгоритм побудови МПД. \*
44. Алгоритми Крускала і Прима побудови мінімального покриваючого дерева. \*
45. Алгоритм Борувки побудови мінімального покриваючого дерева.
46. Щільнісний алгоритм кластеризації MeanShift.\*
47. Щільнісний алгоритм кластеризації DBSCAN.\*
48. Ідея базового алгоритму FOREL.
49. Етапи базового алгоритму FOREL. Переваги і недоліки.
50. Ідея та етапи FOREL-3.
51. Алгоритм FOREL-4.
52. Вибір кількості кластерів.
53. Метрики якості кластеризації: Homogeneity, Completeness, V-measure, Silhouette Coefficient. \*
54. Метрики якості кластеризації: Adjusted Rand Index, Adjusted Mutual Information.

55. Метрики якості кластеризації: Calinski-Harabasz Index, Davies-Bouldin index.
56. Постановка задачі кластеризації на основі мережі Кохонена. Опис мережі Кохонена. \*
57. Алгоритм стохастичного градієнта для розрахунку центрів кластерів в мережі Кохонена.\*
58. Поняття карти Кохонена. Метрики на карті. Навести ілюстрацію карти.\*
59. Алгоритм навчання карти Кохонена.
60. Візуалізація карт Кохонена, типи карт. Переваги і недоліки методу карт Кохонена.
61. Означення нейронної мережі. Модель МакКаллока-Піттса. \*
62. Функції активації в нейронних мережах: порогові, сигмоїдальна, гіперболічний тангенс. Їх властивості.
63. Побудова ансамблю моделей методом бустингу. Алгоритм AdaBoostClassifier. \*
64. Побудова ансамблю моделей методом бегінгу. Алгоритм BaggingClassifier. \*
65. Алгоритм GradientBoostingClassifier.
66. Побудова ансамблю моделей методом стекінгу. Алгоритм StackingClassifier. \*
67. Побудова ансамблю моделей методом голосування (Voting). Алгоритм VotingClassifier. \*