

Савенко О.С.
2021 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Машинне навчання

Галузь знань 12 – Інформаційні технології
Спеціальність 121 – Інженерія програмного забезпечення
Рівень вищої освіти – Другий магістерський
Освітньо-професійна програма – Інженерія програмного забезпечення
Обсяг дисципліни – 5 кредитів ЄКТС
Шифр дисципліни – ОПП.03
Статус дисципліни: Обов'язкова
Мова навчання – українська
Факультет – Програмування та комп'ютерних і телекомунікаційних систем
Кафедра – Інженерії програмного забезпечення


Форма навчання	Курс	Семестр	Обсяг дисципліни Кредити ЄКТС	Кількість годин						Форма семестрового контролю			
				Аудиторні заняття				са	Самостійна робота, у т.ч. ІРС	Курсовий проект	Курсова робота	Залік	Іспит
				Разом	Лекції	Лабораторні роботи	Практичні заняття						
Д	1 маг	2	5	150	18	36			96				
Разом ДФН			5	150	18	36			96				

Робочу програму складено на основі освітньої програми.

Програма складена  Л.П. Бедратюк
Підпис Ініціали, прізвище

Схвалена на засіданні кафедри Інженерії програмного забезпечення

Протокол від 21 серпня 2021 № 1

Зав. кафедри інженерії програмного забезпечення  Л.П. Бедратюк
Підпис Ініціали, прізвище

Робоча програма розглянута та схвалена Вченою радою факультету програмування та комп'ютерних і телекомунікаційних систем

Голова Вченої ради  О.С. Савенко
Підпис Ініціали, прізвище

МАШИННЕ НАВЧАННЯ

Тип дисципліни	Обов'язкова
Рівень вищої освіти	другий (магістерський)
Мова викладання	Українська
Семестр	2
Кількість встановлених кредитів ЄКТС	5
Форми навчання, для яких викладається дисципліна	Денна

Результати навчання. Відповідно до Стандарту вищої освіти із зазначеної спеціальності та освітньої програми дисципліна має забезпечити: **компетентності:** Здатність генерувати нові ідеї (креативність). Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення. Здатність розробляти і реалізовувати наукові та/або прикладні проекти у сфері інженерії програмного забезпечення. Здатність розвивати і реалізовувати нові конкурентоспроможні ідеї в інженерії програмного забезпечення. Здатність критично осмислювати проблеми у галузі інформаційних технологій та на межі галузей знань, інтегрувати відповідні знання та розв'язувати складні задачі у широких або мультидисциплінарних контекстах. Здатність виконувати проектування програмних систем та їх складових компонентів із використанням поглиблених знань з оптимізації, реінжинірингу програмних систем, управління проектами та якістю з вибором парадигм програмування та застосування машинного навчання.

програми результати навчання: Будувати і досліджувати моделі інформаційних процесів у прикладній області. Виявляти інформаційні потреби і класифікувати дані для проектування програмного забезпечення. Прогнозувати розвиток програмних систем та інформаційних технологій. Вміти використовувати методи фундаментальних і прикладних дисциплін інженерії програмного забезпечення при проектуванні архітектури та розробці програмних систем із використанням поглиблених знань з оптимізації, реінжинірингу програмних систем, управління проектами та якістю з вибором парадигм програмування та застосування машинного навчання

Зміст навчальної дисципліни. Предмет і завдання машинного навчання і аналізу даних. Основні принципи, завдання та підходи, використання в різних областях науки і індустрії. Основні етапи еволюції алгоритмів машинного навчання. Бібліотеки Python для машинного навчання. Метричні класифікатори Загальний вигляд метричного класифікатора. Алгоритм К найближчих сусідів. Алгоритми відбору еталонів. Алгоритми кластеризації Алгоритми кластеризації з фіксованою кількістю кластерів. Алгоритми кластеризації по щільності. Ієрархічна кластеризація. Деревя рішень . Критерій інформаційного виграшу і критерій Джині. Лінійні класифікатори. Перцептрон і розділююча гіперплощина. Перехід в простір підвищеної розмірності. Метод опорних векторів. Логістична регресія. Метод найшвидшого спуску. Нейронні мережі і глибоке навчання. Нейронні мережі та алгоритм зворотного поширення градієнта. Нейронні мережі для обробки зображень, природної мови. Змагально-генеративні нейронні мережі.

Запланована навчальна діяльність. лекції – 18 год., лабораторні роботи – 36 год., самостійна робота – 96 год., разом – 150 год.

Методи навчання: Методи проблемного викладання, словесні, наочні(лекції); пояснювально-ілюстративні, частково-пошукові, лабораторна робота, самостійна робота: індивідуальні завдання.

Форми і методи оцінювання результатів навчання: усне опитування, захисти лабораторних робіт, письмовий іспит

Вид семестрового контролю: іспит

Навчальні ресурси:

1. S. Raschka, V. Mirjalili, Python Machine Learning - Second Edition: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow, Packt, 2017.
2. Chollet F., Deep Learning with Python, Manning Publications Co., 2018, 386 pp
3. I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016
4. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006, -749p
5. Модульне середовище для навчання MOODLE. Доступ до ресурсу: <https://msn.khnu.km.ua>.
6. Електронна бібліотека університету. Доступ до ресурсу: http://lib.khnu.km.ua/asp/php_f/plage_lib.php.

Викладач: доктор фізико-математичних наук, професор Бедратюк Л.П.

3. ПОЯСНЮВАЛЬНА ЗАПИСКА

Дисципліна "Машинне навчання" є дисципліною професійної підготовки.

Мета дисципліни – Метою дисципліни «Машинне навчання» є: 1) знати ключові поняття, цілі та завдання машинного навчання; методологічні основи застосування алгоритмів машинного навчання. 2) ознайомити студентів з сучасними парадигмами та технологіями машинного навчання. 3) підготувати студентів до застосування технологій машинного навчання у складних і непередбачуваних умовах, що потребує застосування нових підходів та породження нових ідей (креативності) 4) ознайомити студентів з основними типами задач машинного навчання. 5) навчити візуалізувати результати роботи алгоритмів машинного навчання, ефективно вибирати метод машинного навчання, відповідний поставленій задачі, інтерпретувати отримані результати.

Предмет дисципліни – комплекс питань, пов'язаних з теорією, принципами, технологіями та методами машинного навчання.

Завдання дисципліни. сформувати цілісне уявлення про методи машинного навчання, про обмеження цих методів і про прикладні задачі, які можна вирішувати з їх допомогою. Надати студентам теоретичні знання і практичні навички розробки програмного забезпечення з використанням мови Python та бібліотек Keras, Open CV, а також навичок проектування, розробки архітектури та навчання штучних нейронних мереж а також оцінювати їхню ефективність; навчитися формулювати професійні завдання на мові машинного навчання, створювати ефективні програми для вирішення завдань машинного навчання; володіти навичками автоматичного навчання створених моделей і застосовувати ці моделі для вирішення прикладних задач

Відповідно до Стандарту вищої освіти із зазначеної спеціальності та освітньої програми дисципліна має забезпечити:

компетентності:

ЗК-5 Здатність генерувати нові ідеї (креативність).ФК-1. Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення. ФК-2. Здатність розробляти і реалізовувати наукові та/або прикладні проєкти у сфері інженерії програмного забезпечення. ФК-4. Здатність розвивати і реалізовувати нові конкурентоспроможні ідеї в інженерії програмного забезпечення. ФК-7. Здатність критично осмислювати проблеми у галузі інформаційних технологій та на межі галузей знань, інтегрувати відповідні знання та розв'язувати складні задачі у широких або мультидисциплінарних контекстах. ФК-10 Здатність виконувати проектування програмних систем та їх складових компонентів із використанням поглиблених знань з оптимізації, реінжинірингу програмних систем, управління проєктами та якістю з вибором парадигм програмування та застосування машинного навчання

програмні результати навчання: ПРН-3 Будувати і досліджувати моделі інформаційних процесів у прикладній області. ПРН-4 Виявляти інформаційні потреби і класифікувати дані для проектування програмного забезпечення. ПРН-14 Прогнозувати розвиток програмних систем та інформаційних технологій. ПРН-18 Вміти використовувати методи фундаментальних і прикладних дисциплін інженерії програмного забезпечення при проектуванні архітектури та розробці програмних систем із використанням поглиблених знань з оптимізації, реінжинірингу програмних систем, управління проєктами та якістю з вибором парадигм програмування та застосування машинного навчання

4. СТРУКТУРА ЗАЛКОВИХ КРЕДИТІВ ДИСЦИПЛІНИ

Назва теми	Кількість годин, відведених на:			
	лекції	лабор. роботи	СРС	Разом
<i>Другий семестр</i>				
Тема 1. Основи машинного навчання	12	20	50	82
Тема 2. Глибоке навчання	6	16	46	68
Разом за семестр	18	36	96	150

5. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

5.1. Зміст лекційного курсу*

Номер лекції	Перелік тем лекцій, їх анотації	Кількість годин
	<i>Другий семестр</i>	
1	<p>Тема 1. Основи машинного навчання <i>Лекція 1. Вступ до машинного навчання</i> Формальна постановка задачі машинного навчання. Об'єкти та ознаки. Типи задач машинного навчання- задачі класифікації (розпізнавання образів), відновлення регресії, прогнозування, ранжування. Задачі навчання з учителем, навчання без учителя, навчання з підсиленням. Моделі алгоритмів. Лінійна модель. Приклад: Задача класифікації ірисів. Поліноміальна модель. Оцінка моделі - функціонал якості, функція втрат. Перенавчання та недонавчання. Приклади задач машинного навчання. Експерименти на реальних та синтетичних даних. Міжгалузевий стандарт CRISP-DM для вирішення завдань інтелектуального аналізу даних. Життєвий цикл дослідження даних. Загальний алгоритм розв'язання задач машинного навчання. Основні етапи еволюції алгоритмів машинного навчання. Переваги машинного навчання над традиційними методами. Майбутнє машинного навчання. Літ.: [2,3,6]</p>	2
2	<p>Тема 1. Основи машинного навчання <i>Лекція 2. Метричні та лінійні методи машинного навчання.</i> Метричні методи класифікації - гіпотеза компактності і неперервності, формалізація поняття близькості, узагальнений метричний класифікатор. Метод найближчих k сусідів (kNN), переваги та недоліки, метод вікна Парзена. Метричні методи в задачі відновлення регресії - задачі регресії і метод найменших квадратів, непараметрична регресія. Проблема виявлення та згладжування викидів. Лінійні методи класифікації - лінійні моделі та області застосування лінійних моделей. Вимірювання помилки в задачі регресії. Навчання лінійної регресії.. Градієнтний спуск і оцінювання градієнта. Модифікації градієнтного спуску. Алгоритм SG (стохастичний градієнт). Алгоритм SAG (стохастичний усереднений градієнт). Мультиколінеарність, проблема перенавчання та методи її вирішення, регуляризація. Літ.: [3,6,12]</p>	2
		2

3	<p>Тема 1. Основи машинного навчання <i>Лекція 3. Метод опорних векторів (SVM) і логістична регресія.</i> Навчання лінійного класифікатора, метрики якості класифікації - точність, повнота, частка правильних відповідей, матриця помилок, площа під кривою, індекс Джині. Апроксимація і регуляризація емпіричного ризику. Оптимальна розділююча гіперплощина. Випадок лінійно-нероздільної вибірки. Переваги та недоліки методу опорних векторів. Узагальнення на нелінійний випадок. Логістична регресія та побудова лінійних класифікаторів, оптимізація параметрів логістичної регресії. Літ.: [3,6,12]</p>	
4	<p>Тема 1. Основи машинного навчання <i>Лекція 4. Лінійна регресія, зниження розмірності(узагальнення) та метод головних компонент.</i> Багатовимірна лінійна регресія. сингулярний розклад матриці (singular-value decomposition, SVD). Оптимізація методу найменших квадратів та сингулярний розклад. Проблема мультиколінеарності та перенавчання. Хребтова регресія (ridge regression), метод LASSO. Метод головних компонент (principal component analysis), основна теорема, зв'язок з сингулярним розкладом. Ефективна розмірність вибірки. Застосування PCA для задачі багатовимірної класифікації методом найменших квадратів на нових ознаках. Літ.: [3,6,12]</p>	2
5	<p>Тема 1. Основи машинного навчання <i>Лекція 5. Дерева рішень.</i> Побудова дерева рішень, критерій інформативності, регресія, класифікація та помилки класифікації, критерій Джині, ентропійний критерій. Критерії зупинки - Правила зупинки розбиття дерева. Обрізання дерева. Алгоритм ID3. Перенавчання дерев рішень. Обробка неперервних атрибутів. Навчання на даних з пропусками. Реалізація дерев в бібліотеках Python. Бегінг ((Bagging), випадкові ліси і розклад помилки на зміщення і варіацію. Бутстрап. Бегінг(bagging, bootstrap aggregation). Випадкові ліси, зв'язок з матричними методами. Літ.: [5,6,7]</p>	2
6	<p>Тема 1. Основи машинного навчання <i>Лекція 6. Байєсова класифікація.</i> Умовна ймовірність. Формула повної ймовірності. Формула Байєса. Статистичне розпізнавання образів. Наївний байєсівський класифікатор. Завдання класифікації спаму. Критерій відношення правдоподібності. Байєсівський рівень помилки. Байєсівський ризик. Критерій Байєса. Максимальний апостеріорного критерій. Критерій максимальної правдоподібності. Байєсовські класифікатори для багатьох (більше 2) класів. Байєсовські класифікатори для нормально розподілених класів за умови різної структури матриці коваріації Літ.: [5,6,7]</p>	2
7	<p>Тема 2. Глибоке навчання <i>Лекція 7. Нейронні мережі.</i> Нейронні мережі та глибоке навчання. <i>Робота в Python фреймворк Keras.</i> Анатомія нейронної мережі: Шари - будівельні блоки глибокого навчання; Моделі: мережі шарів; Функції втрат та оптимізатори. Вступ до Keras: Keras, TensorFlow, Theano і CNTK.Розробка з використанням Keras. Згорткові нейронні мережі: Операція згортки; Завантаження даних; Конструювання мереж; Попередня обробка даних; Використання попередньо навченої згорткової нейронної мережі: Виділення ознак; Донавчання. Візуалізація знань, згорткової нейронної мережі Глибокі згорткові нейронні мережі. Розпізнавання та аналіз зображень при допомозі</p>	2

	згорткових нейронних мереж. Моменти зображення, Геометричні моменти в 2D. TRS-інваріанти від геометричних моментів: Інваріанти відносно зміщення; Інваріанти до рівномірного масштабування; Інваріанти до нерівномірного масштабування; Традиційні інваріанти обертання. Інваріанти обертання, що використовують кругові моменти. Інваріанти обертання від складних моментів. Літ.: [1,8,13,14]	
8	Тема 2. Глибоке навчання. Лекція 8. Рекурентні нейронні мережі та обробка текстових даних. Глибоке навчання для тексту і послідовностей. Робота з текстовими даними - Пряме кодування слів і символів. Векторне подання слів Рекурентні нейронні мережі- рекурентні шари в Keras. Шари LSTM і GRU Приклад використання шару LSTM з Keras. Покращені методи використання рекурентних нейронних мереж. Завдання прогнозування температури. Підготовка даних. Базове рішення без залучення машинного навчання Порівняння рішення с залученням машинного навчання та без нього. Перше базове рекурентне рішення. Використання рекурентного проріджування для боротьби з перенавчанням. Накладення декількох рекурентних шарів один на одного. Використання двонапрямлених рекурентних нейронних мереж. Обробка послідовностей за допомогою згорткових нейронних мереж- Обробка послідовних даних за допомогою одновимірної згорткової нейронної мережі, Вибір сусідніх значень в одновимірній послідовності даних. Реалізація одновимірної згорткової мережі. Об'єднання згорткових і рекурентних мереж для обробки довгих послідовностей Літ.: [1,8,11,14,16]	2
9	Тема 2. Глибоке навчання. Лекція 9. Змагально-генеративні нейронні мережі (GAN). Генеративне і дискримінантне моделювання. Поява генеративного моделювання. Базові принципи генеративного моделювання. Імовірнісні генеративні моделі. Наївна байєсівська параметрична модель. Навчання подання. Налаштування оточення. Варіаційні автокодувальники. Аналіз автокодувальника. Використання варіаційного автокодувальника для генерації зображень облич. Навчання VAE. Перетворення однієї обличчя в інше. Генеративно-змагальні мережі. Дискримінатор. Генератор. Навчання генеративно-змагальної мережі. Проблеми генеративно-змагальних мереж - Коливання втрат, Колапс моделі, Неінформативні втрати, Гіперпараметри. Генеративно-змагальні мережі з функцією втрат Вассерштейна. Нейронна перенесення стилю, мережа CycleGAN. Реалізація GAN в Keras і PyTorch. Літ.: [1,8,9]	2
	Разом за другий семестр:	18

5.2 Зміст лабораторних занять

Перелік лабораторних занять

№ з/п	Тема лабораторного заняття	Кількість годин
<i>Другий семестр</i>		
1	Встановлення Python, робочого середовища Jupyter. Обробка даних в Python-бібліотеці Pandas. Літ.: [1]	4
2	Лінійна регресія Літ.: [1,4]	4
3	Класифікація методом k-найближчих сусідів. Літ.: [1,4,6]	4
4	Дерево рішень Літ.: [6];	4
5	Проектування та навчання штучної нейронної мережі для розпізнавання простих зображень Літ.: [1,11]	4
6	Проектування та навчання згорткової штучної нейронної мережі для задач аналізу зображень Літ.: [1,11]	4
7	Побудова та використання згорткової нейронної мережі для класифікація великих зображень [1,11]	6
8	Рекурентні нейронні мережі і обробка текстів Літ.: [1,5,16]	4
9	Підсумкове заняття	2
Разом за другий семестр:		36

5.3 Зміст самостійної (у т.ч. індивідуальної) роботи Зміст самостійної роботи студентів денної форми навчання

Номер тижня	Зміст самостійної роботи	К-ть годин
<i>Перший семестр</i>		
1-2	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття.	10
3-4	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття.	10
5-6	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття.	10
7-8	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття.	10
9-10	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття.	10
11-12	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття.	10
13-14	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття.	10
15-16	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття.	10
17-18	Опрацювання лекційного матеріалу. Підготовка до підсумкового контролю.	16
Разом за 2-ий семестр		96

6. ТЕХНОЛОГІЇ НАВЧАННЯ

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних методів: Методи проблемного викладання, словесні, наочні(лекції); пояснювально-ілюстративні, частково-пошукові, лабораторна робота, самостійна робота: індивідуальні завдання. Всі заняття проводяться з використанням інформаційних технологій і мають за мету – сформувані цілісне уявлення про методи машинного навчання, про обмеження цих методів і про прикладні задачі, які можна вирішувати з їх допомогою

7. ФОРМИ І МЕТОДИ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Поточний контроль здійснюється під час лекційних та лабораторних занять. Семестровий контроль проводиться у формі іспиту. При виведенні остаточної оцінки враховуються результати поточного контролю та письмового іспиту.

Процес оцінювання підготовленості студента можна розділити на етапи:

Перший етап оцінювання направлений на визначення знань інформаційного мінімуму. Якщо студент твердо засвоїв визначену навчальним планом суму формальних знань, то це означає, що він вміє використати їх при вирішенні різних питань з безпеки та захисту комп'ютерних систем, проектування та реалізації програмних систем захисту інформації та їх компонентів, вміє розширити їх. При цьому необхідно встановити рівні та критерії сформованості знань щодо змісту навчальних елементів. Такими рівнями є:

Ознайомчо-орієнтовний (ОО) – особа має орієнтовне уявлення щодо понять, які вивчаються, здатна: алгоритмізувати та програмувати основні елементи програмних систем захисту інформації та їх компонентів мовами програмування, обирати сучасні технології розробки програмного забезпечення елементарного призначення, обґрунтовано використовувати сучасні середовища розроблення програмного забезпечення для розроблення програм, вміти налагоджувати існуючі засоби для забезпечення безпеки комп'ютерних систем.

Понятійно-аналітичний (ПА) – особа має чітке уявлення щодо навчального об'єкту, здатна перенести раніше засвоєні знання на типові ситуації.

Продуктивно-синтетичний (ПС) – особа має глибоке розуміння щодо навчального об'єкту, здатна здійснювати синтез, генерувати нові ідеї та уявлення, переносити раніше засвоєні знання на нетипові, нестандартні ситуації.

При викладанні дисципліни використовуються такі види навчальних занять, як лекції, лабораторні роботи, індивідуальне консультування і керівництво самостійною роботою студента.

Кожний вид роботи з дисципліни оцінюється за *чотирибальною* шкалою. Семестрова підсумкова оцінка визначається як середньозважена з усіх видів навчальної роботи, виконаних і зданих *позитивно* з врахуванням коефіцієнта вагомості. Вагові коефіцієнти змінюються залежно від структури дисципліни і важливості окремих її видів робіт. Студент, який набрав позитивний середньозважений бал за поточну роботу і не здав підсумковий контрольний захід (іспит), вважається невстигаючим.

При оцінюванні знань студентів використовуються різні засоби контролю, зокрема: усне опитування перед допуском до виконання лабораторної роботи – здійснюється на її початку; якість виконання, набуття теоретичних знань і практичних навичок перевіряється шляхом захисту кожної лабораторної роботи згідно з робочою програмою дисципліни.

Оцінка, яка виставляється за *лабораторне заняття*, складається з таких елементів: усне опитування студентів перед допуском до виконання лабораторної роботи; знання теоретичного матеріалу з теми; якість оформлення звіту і графічної частини; вміння студента обґрунтувати прийняті конструктивні рішення; своєчасний захист лабораторної роботи. Для виконання програми дисципліни студент повинен отримати 8 позитивних оцінок за лабораторні роботи в семестрі.

Термін захисту лабораторної роботи вважається своєчасним, якщо студент захистив її на наступному після виконання роботи занятті. За несвоєчасний захист лабораторної роботи з неповажної причини студент за позитивну відповідь отримує оцінку «задовільно».

Пропущене лабораторне заняття студент повинен відпрацювати в лабораторіях кафедри у встановлений викладачем термін з реєстрацією у відповідному журналі кафедри, але не пізніше,

ніж за тиждень до завершення теоретичних занять у семестрі.

При оцінюванні знань студентів викладач керується такими критеріями.

Оцінку „відмінно”, за шкалою ECTS – А (див. шкалу оцінок), отримує студент за глибоке і повне опанування змісту навчального матеріалу, в якому він легко орієнтується, понятійного апарату, за уміння зв’язувати теорію з практикою, вирішувати практичні завдання, висловлювати і обґрунтовувати свої судження. Відмінна оцінка передбачає грамотний, логічний виклад відповіді (як в усній, так і в письмовій формі), якісне зовнішнє оформлення. Студент повинен набути практичних навичок із складання різних алгоритмів та розробки програм за цими алгоритмами. Оцінка "відмінно" виставляється студенту, який глибоко засвоїв предметну область та вміє застосовувати її на практиці. Студент не повинен вагатися при видозміні запитання, повинен робити детальні та узагальнюючі висновки.

Оцінку „добре”, за шкалою ECTS – В, отримує студент за повне засвоєння навчального матеріалу, володіння понятійним апаратом, орієнтування в вивченому матеріалі, свідоме використання знань для вирішення практичних завдань, грамотний виклад відповіді, але у змісті і формі відповіді мали місце окремі неточності (похибки), нечіткі формулювання закономірностей тощо. Відповідь студента повинна будуватись на основі самостійного мислення.

Оцінку „добре”, за шкалою ECTS – С, отримує студент за правильну відповідь з однією суттєвою помилкою.

Оцінки "задовільно", за шкалою ECTS – D, заслуговує студент, який виявив знання основного навчально-програмного матеріалу в обсязі, необхідному для подальшого навчання та практичної діяльності за професією, що справляється з виконанням практичних завдань, передбачених програмою. Як правило, відповідь студента будується на рівні репродуктивного мислення, студент слабо знає структуру курсу, допускає помилки у відповіді, засвоїв і набув практичних навичок, але допустив неточності. Вагається при відповіді на видозмінене запитання, разом з тим студент володіє знаннями, що дозволяють йому під керівництвом викладача усунути неточності у відповіді.

Оцінки "задовільно", за шкалою ECTS – E, заслуговує студент за неповне опанування програмного матеріалу, але отримані знання і набуті практичні навички.

Оцінка „незадовільно”, за шкалою ECTS – FХ, виставляється, коли студент має розрізнені, безсистемні знання, не вміє виділяти головне і другорядне, допускається помилок у визначенні понять, перекручує їх зміст, хаотично і невпевнено викладає матеріал, не може використовувати знання при вирішенні практичних завдань. Як правило, оцінка "незадовільно" виставляється студенту, який не може продовжити навчання без додаткових знань з курсу.

Оцінка „незадовільно”, за шкалою ECTS – F, виставляється студенту за повне незнання і нерозуміння навчального матеріалу або відмову від відповіді і передбачає повторне навчання студента з дисципліни.

Кожний вид роботи оцінюється за чотирибальною шкалою. Семестрова підсумкова оцінка визначається як середньозважена з усіх видів робіт

Структурування дисципліни за видами робіт і оцінювання результатів навчання студентів денної форми навчання у семестрі за ваговими коефіцієнтами

Аудиторна робота								Самостійна, індивідуальна робота		Іспит
<i>2 семестр</i>										
Лабораторні роботи №:								Контроль:		Іспит
1	2	3	4	5	6	7	8	ТК Т 1-4	ТК Т 5-8	
ВК:								0,6		0,4

Умовні позначення: ТК – тестовий контроль; ВК – ваговий коефіцієнт

Підсумкова семестрова оцінка за національною шкалою і шкалою ECTS встановлюється в автоматизованому режимі після внесення усіх оцінок до електронного журналу. При цьому за вітчизняною шкалою ставиться «зараховано», а за шкалою ECTS – буквене позначення оцінки, що відповідає набраній студентом кількості балів.

Співвідношення вітчизняної шкали оцінювання і шкали оцінювання ECTS

Оцінка ЄКТС	Інтервальна шкала балів	Вітчизняна оцінка	
A	4,75–5,00	5	<i>Відмінно</i> – глибоке і повне опанування навчального матеріалу і виявлення відповідних умінь та навиків
B	4,25–4,74	4	<i>Добре</i> – повне знання навчального матеріалу з кількома незначними помилками
C	3,75–4,24	4	<i>Добре</i> – в загальному правильна відповідь з двома-трьома суттєвими помилками
D	3,25–3,74	3	<i>Задовільно</i> – неповне опанування програмного матеріалу, але достатнє для практичної діяльності за професією
E	3,00–3,24	3	<i>Задовільно</i> – неповне опанування програмного матеріалу, що задовольняє мінімальні критерії оцінювання
FX	2,00–2,99	2	<i>Незадовільно</i> – безсистемність одержаних знань і неможливість продовжити навчання без додаткових знань з дисципліни
F	0,00–1,99	2	<i>Незадовільно</i> – необхідна серйозна подальша робота і повторне вивчення дисципліни

8. ПИТАННЯ ДЛЯ САМОКОНТРОЛЮ ЗДОБУТИХ СТУДЕНТАМИ ЗНАТЬ

1. Формальна постановка задачі машинного навчання.
2. Об'єкти та ознаки.
3. Типи задач машинного навчання- задачі класифікації (розпізнавання образів), відновлення регресії, прогнозування, ранжування.
4. Задачі навчання з учителем, навчання без учителя, навчання з підсилення.
5. Моделі алгоритмів. Лінійна модель. Поліноміальна модель.
6. Оцінка моделі - функціонал якості, функція втрат. Переневчання та недонавчання.
7. Приклади задач машинного навчання.
8. Експерименти на реальних та синтетичних даних.
9. Міжгалузевий стандарт CRISP-DM для вирішення завдань інтелектуального аналізу даних.
10. Життєвий цикл дослідження даних.
11. Алгоритм вирішення задачі машинного навчання.
12. Метричні методи класифікації - гіпотеза компактності і неперервності, формалізація поняття близькості, узагальнений метричний класифікатор.
13. Метод найближчих k сусідів (kNN), переваги та недоліки, метод вікна Парзена.
14. Метричні методи в задачі відновлення регресії - задачі регресії і метод найменших квадратів, непараметрична регресія.
15. Проблема виявлення та згладжування викидів
16. Лінійні методи класифікації - лінійні моделі та області застосування лінійних моделей.
17. Вимірювання помилки в задачі регресії. Навчання лінійної регресії.
18. Градієнтний спуск і оцінювання градієнта. Модифікації градієнтного спуску.
19. Алгоритм SG (стохастичний градієнт).
20. Алгоритм SAG (стохастичний усереднений градієнт).
21. Мультиколінеарність, проблема перенавчання та методи її вирішення, регуляризація.
22. Метод опорних векторів (SVM) і логістична регресія.
23. Навчання лінійного класифікатора, метрики якості класифікації - точність, повнота, частка правильних відповідей, матриця помилок, площа під кривою, індекс Джині.
24. Апроксимація і регуляризація емпіричного ризику.
25. Оптимальна розділююча гіперплощина.
26. Випадок лінійно-нероздільної вибірки.
27. Переваги та недоліки методу опорних векторів.
28. Узагальнення на нелінійний випадок.
29. Логістична регресія та побудова лінійних класифікаторів, оптимізація параметрів логістичної регресії.

30. Лінійна регресія, зниження розмірності(узагальнення) та метод головних компонент.
31. Багатовимірна лінійна регресія. сингулярний розклад матриці (singular-value decomposition, SVD).
32. Оптимізація методу найменших квадратів та сингулярний розклад.
33. Проблема мультиколінеарності та перенавчання.
34. Хребтова регресія (ridge regression), метод LASSO.
35. Метод головних компонент (principal component analysis), основна теорема, зв'язок з сингулярним розкладом.
36. Застосування PCA для задачі багатовимірної класифікації методом найменших квадратів на нових ознаках.
37. Дерева рішень. Побудова дерева рішень, критерій інформативності, регресія, класифікація та помилки класифікації, критерій Джині, ентропійний критерій.
38. Критерії зупинки - Правила зупинки розбиття дерева. Обрізання дерева.
39. Алгоритм ID3.
40. Перенавчання дерев рішень. Обробка неперервних атрибутів.
41. Навчання на даних з пропусками. Реалізація дерев в бібліотеках Python.
42. Бегінг ((Bagging), випадкові ліси і розклад помилки на зміщення і варіацію.
43. Бутстрап. Бегінг(bagging, bootstrap aggregation).
44. Випадкові ліси, зв'язок з матричними методами.
45. Байєсова класифікація.
46. Статистичне розпізнавання образів. Наївний байєсівський класифікатор.
47. Завдання класифікації спаму. Критерій відношення правдоподібності.
48. Критерій максимальної правдоподібності. Байєсовські класифікатори для багатьох (більше 2) класів.
49. Байєсовські класифікатори для нормально розподілених класів за умови різної структури матриці коваріації
50. Нейронні мережі. Нейронні мережі та глибоке навчання.
51. Робота в Python фреймворк Keras.
52. Анатомія нейронної мережі: Шари - будівельні блоки глибокого навчання; Моделі: мережі шарів;
53. Функції втрат та оптимізатори.
54. Вступ до Keras: Keras, TensorFlow, Theano і CNTK.
55. Згорткові нейронні мережі: Операція згортки;
56. Завантаження даних; Конструювання мереж; Попередня обробка даних;
57. Використання попередньо навченої згорткової нейронної мережі: Виділення ознак; Донавчання.
58. Візуалізація знань, згорткової нейронної мережі
59. Глибокі згорткові нейронні мережі.
60. Розпізнавання та аналіз зображень при допомозі згорткових нейронних мереж.
61. Моменти зображення, Геометричні моменти в 2D.
62. TRS-інваріанти від геометричних моментів: Інваріанти відносно зміщення; Інваріанти до рівномірного масштабування; Інваріанти до нерівномірного масштабування;
63. Глибоке навчання для тексту і послідовностей.
64. Робота з текстовими даними - Пряме кодування слів і символів. Векторне подання слів
65. Рекурентні нейронні мережі- рекурентні шари в Keras.
66. Шари LSTM і GRU Приклад використання шару LSTM з Keras.
67. Покращені методи використання рекурентних нейронних мереж.
68. Завдання прогнозування температури. Підготовка даних.
69. Використання рекурентного проріджування для боротьби з перенавчанням. Накладення декількох рекурентних шарів один на одного. Використання двонапрямлених рекурентних нейронних мереж.
70. Обробка послідовностей за допомогою згорткових нейронних мереж-
71. Обробка послідовних даних за допомогою одновимірної згорткової нейронної мережі
72. Вибір сусідніх значень в одновимірній послідовності даних.
73. Реалізація одновимірної згорткової мережі. Об'єднання згорткових рекурентних мереж для обробки довгих послідовностей

74. Змагально-генеративні нейронні мережі (GAN). Генеративне і дискримінантне моделювання. Поява генеративного моделювання.
75. Базові принципи генеративного моделювання. Імовірнісні генеративні моделі. Наївна байєсівська параметрична модель.

9. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

Навчальний процес з дисципліни «Методології та технології розробки програмних систем» повністю і в достатній кількості забезпечений необхідною навчально-методичною літературою.

10. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна

1. Chollet F., Deep Learning with Python, Manning Publications Co., 2018, 386 pp
2. Shalev-Shwartz S., Ben-David S., Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press, 2014,-416
3. Mohri Mehryar, Afshin Rostamizadeh, Ameet Talwalkar, Foundations of Machine Learning, The MIT Press, 2018,-505p
4. Andreas C. Müller, Sarah Guido, Introduction to Machine Learning with Python: A Guide for Data Scientists, O'Reilly Media, 2016,-392p
5. Puneet Mathur, Machine Learning Applications Using Python - Cases Studies from Healthcare, Retail, and Finance, Apress, 2019, -379p
6. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006, -749p
7. Mohri M., Rostamizadeh A., Talwalkar A. Foundations of Machine Learning. MIT Press, 2012.
8. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, The MIT Press, —2016, 800p
9. Jakub Langr, Vladimir Bok, GANs in Action: Deep learning with Generative Adversarial Networks, Manning Publications, 2019,-276
10. David Foster, Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play, O'Reilly Media, 2019,-330p
11. Dawson-Howe K., A practical introduction to computer vision with OpenCV, Wiley, 2014,-235p.
12. Hastie T., Tibshirani R., Friedman J., The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, 2017, 768p..
13. Liu H., Cocea M. Granular Computing based Machine Learning. A Big Data Processing Approach / H. Liu, M. Cocea. — Springer, 2018. — 118 p.
14. Laros D.T., Laros C.D. Discovering knowledge in data/ Daniel T. Laros and Chantal D. Laros – Wiley&Sons: New Jersey, 2014. — 332 p
15. Brownlee J. Machine learning mastery with Python /J. Brownlee. — Machine Learning Mastery, 2016. — 179 p.
16. Delip Rao and Brian McMahan, Natural Language Processing with PyTorch. Build Intelligent Language Applications Using Deep Learning, 2019 , -256p

Допоміжна

1. Shih F., Image Processing and Pattern Recognition, Fundamentals and Techniques, Wiley, 552p
2. John Hany, Greg Walters, Implement next-generation neural networks to build powerful GAN models using Python, Packt Publishing, 2019, 301p
3. Sridevi P., Ravishankar C., Image processing and acquisition using Python, Chapman and Hall/CRC, 2015, 372pp
4. Fernández V. Mastering OpenCV 4 with Python : A Practical Guide Covering Topics from Image Processing, Augmented Reality to Deep Learning with OpenCV 4 and Python 3. 7., Packt Publishing, 2019, 617pp
5. Pajankar A., Python 3 Image Processing: Learn Image Processing with Python 3, NumPy, Matplotlib, and Scikit-image, Packt Publishing, 2019, 752 pp
6. Garsia B., Learning Image Processing with OpenCV, Packt Publishing, 2015, -319c

7. Howse J., Joshi P., Beyeler M., Opencv: Computer Vision Projects with Python, Packt Publishing, 2016, 570pp
8. Mohammed J. Zaki, Wagner Meira Jr. Data Mining and Analysis. Fundamental Concepts and Algorithms. Cambridge University Press, 2014
9. den Bakker I., Python Deep Learning Cookbook, Packt P., 2017, 436pp
10. Rosebrock A., Deep Learning for Computer Vision with Python, Pyimagesearch, 2017, 210 pp
11. Sandipan Dey, Hands-On Image Processing with Python, Packt Publishing Ltd, 2018, 435pp
12. A Krizhevsky, I Sutskever, GE Hinton, Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems, 2012, 1097-1105
13. Diao, L., Peng, J., Dong, J., Kong, F.: Moment invariants under similarity transformation. Pattern Recogn. 48, 3641–3651 (2015)
14. S. Ali Amirshahi, M. Pedersen, and S. X. Yu. Image quality assessment by comparing CNN features between images. Electronic Imaging, 2017(12):42–51, 2017.
15. Рашкевич Ю.М., Ткаченко Р.О., Цмоць Г.І., Пелешко Д.Д. Нейроподібні методи, алгоритми та структури обробки зображень у реальному часі: монографія — Львів: Вид-во Львів. політехніки, 2014. — 256 с.
16. Bedratyuk L., 2D moment invariants from the point of view of the classical invariant theory, Journal of Mathematical Imaging and Vision (2020) 62:1062–1075

11. ІНФОРМАЦІЙНІ РЕСУРСИ

1. Модульне середовище для навчання. Доступ до ресурсу: <https://msn.khnu.km.ua>.
2. Електронна бібліотека університету . Доступ до ресурсу: http://lib.khnu.km.ua/asp/php_f/p1age_lib.php.
3. Репозитарій ХНУ. Доступ до ресурсу: <http://elar.khnu.km.ua/jspui/?locale=uk>.
4. Kaggle Datasets <https://www.kaggle.com/>
5. The International Machine Learning Society. <http://www.machinelearning.org/>
6. MIT <https://openlearninglibrary.mit.edu/>

