Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Кафедра інженерії
програмного забезпечення
в енергетиці

Баєсівська статистика з Python

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни	
Рівень вищої освіти	***Другий (магістерський)***
Галузь знань	*Інформаційні технології*
Спеціальність	*121 Інженерія програмного забезпечення*
Освітня програма	*Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці*
Статус дисципліни	*Вибіркова*
Форма навчання	*Очна (денна)*
Рік підготовки, семестр	*1-й курс, 2-й семестр*
Обсяг дисципліни	*4 кредитів / 120 год. (очна форма: лекцій 36 год., лаб. 18 год., СРС 66 год.)*
Семестровий контроль/ контрольні заходи	*Залік, модульна контрольна робота*
Розклад занять	*http://schedule.kpi.ua/*
Мова викладання	*Українська/Англійська*
Інформація про керівника курсу / викладачів	Лектор: *д. е. н., професор Сігайов Андрій Олександрович* Практичні / Семінарські: Лабораторні: *Сігайов А. О.*
Розміщення курсу	

Програма навчальної дисципліни

1 Опис навчальної дисципліни, її мета, предмет вивчання та результати навчання

Чому майбутньому фахівцю варто вчити саме цю дисципліну?

Баєсівські методи статистичного виведення є повністю природними та надзвичайно потужними. Проте їхнє обговорення зазвичай переважно покладається на ретельний математичний аналіз і штучні приклади, що робить його недоступним для розуміння будь-кого без ґрунтовної математичної освіти. Цей курс, на відміну від багатьох інших, надає вступ до баєсівського виведення з обчислювальної перспективи, наводячи мости між теорією та практикою: студенти звільняються від незграбних ручних обчислень, результати отримуються за допомогою обчислювальних потужностей.

Цей курс висвітлює баєсівське виведення через ймовірнісне програмування за допомогою потужної мови PyMC та тісно пов'язаних інструментів екосистеми Python: NumPy, SciPy та Matplotlib. За допомогою цього підходу студент отримує ефективні розв'язки шляхом малих інкрементальних кроків, без залучення розгалуженого математичного апарату.

Ми починаємо з введення концепцій, що лежать в основі баєсівського виведення, порівнюючи його з іншими методами та направляючи студента скрізь створення та навчання його першої баєсівської моделі. Далі ми вводимо PyMC через низку докладних прикладів та інтуїтивно зрозумілих пояснень, які були вдосконалені після отримання великої кількості відгуків користувачів. Студент дізнається, як використовувати алгоритм Монте-Карло для

Марковських ланцюгів, обирати підхожі розміри вибірки та пріоритети, працювати з функціями втрат і застосовувати баєсівське виведення у різних галузях від фінансів до маркетингу. Після того, як студент опанує ці способи, він зможе постійно звертатися до цього курсу за робочим кодом PyMC, який знадобиться у майбутніх проєктах.

Мета дисципліни. *Ознайомити студентів з ймовірнісним програмуванням і баєсівським виведенням.*

Предмет дисципліни. *Огляд ймовірнісного програмування, включаючи:*

- *Баєсівське мислення та його практичні наслідки;*
- *як комп'ютери виконують баєсівське виведення;*
- *використання бібліотеки PyMC для баєсівського аналізу;*
- *розробка та налагодження моделей з PyMC;*
- *тестування "якості підгонки" моделі;*
- *як і чому працює метод Монте-Карло для марковських ланцюгів;*
- *використання потужності закону великих чисел;*
- *засвоєння ключових понять, як то кластеризація, збіжність, автокореляція та проріджування;*
- *використання функцій втрат для вимірювання якості оцінки залежно від ваших цілей і бажаних результатів;*
- *обрання відповідних апріорних розподілів та розуміння того, як їхній вплив змінюється залежно від розміру дейтасету;*
- *подолання ділеми "дослідження або використання": вирішення, коли "досить гарний" є достатньо гарним*
- *використання баєсівського виведення для покращення A/B-тестування;*
- *розв'язання задач науки про дані, коли наявна тільки мала вибірка*

Очікувані результати навчання.

Фахові компетентності.

ФК 11. Здатність проєктувати та розробляти програмні системи з використанням методів інтелектуального аналізу даних.

Програмні результати навчання.

ПРН 17. Збирати, аналізувати, оцінювати необхідну для розв'язання наукових і прикладних задач інформацію, використовуючи науково-технічну літературу, бази даних та інші джерела.

ПРН 19. Вміти проєктувати та розробляти програмні системи з використанням методів інтелектуального аналізу даних.

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

Дисципліна вивчається у другому семестрі. Пререквізитом є базові знання теорії ймовірностей та мови Python. Постреквізитів у даного курсу на магістерському рівні немає.

3 Зміст навчальної дисципліни

1. *Філософія баєсівського виведення.*
2. *Ще трохи про PyMC.*

3. *Відкриваємо “чорну скриньку” МСМС.*
4. *Найвеличніша з несформульованих теорем.*
5. *Що краще: втратити руку чи ногу?*
6. *Ставимо пріоритети.*
7. *Баєсівське A/B-тестування.*

4 Навчальні матеріали та ресурси

Базова література:

Davidson-Pilon, C. Bayesian Methods for Hackers: Probabilistic Programming and Bayesian Inference: Upper Saddle River, NJ: Addison-Wesley Professional, 2015. 256 с. URL: http://libgen.rs/book/index.php?md5=51d73c88197e1de5d8128f839a61d0e2

Додаткова література:

1. *Kurt, W. Bayesian Statistics the Fun Way: Understanding Statistics and Probability with Star Wars, LEGO, and Rubber Ducks: San Francisco, CA: No Starch Press, 2019. 256 с. URL: http://libgen.rs/book/index.php?md5=BFE1699AAC3B103EDF0FF53169BAA645*
2. *Martin, O. Bayesian Analysis with Python: Introduction to statistical modeling and probabilistic programming using PyMC3 and ArviZ: Birmingham, UK: Packt Publishing, 2018. 356 с. URL: http://libgen.rs/book/index.php?md5=9608A3B42E5D4C1875DCA1C60506B1D2*

Навчальний контент

5 Методика опанування навчальної дисципліни (освітнього компонента)

1. *Філософія баєсівського виведення.*
   - *1.1. Вступ.*
     - *1.1.1. Баєсівське мислення.*
     - *1.1.2. Баєсівське виведення на практиці.*
     - *1.1.3. Чи є коректними частотні методи?*
     - *1.1.4. Проблема великих даних.*
   - *1.2. Понятєвий апарат баєсівського підходу.*
     - *1.2.1. Приклад: підкидання монети (куди ж без нього)*
     - *1.2.2. Приклад: бібліотекар або фермер*
   - *1.3. Розподіли ймовірностей.*
     - *1.3.1. Дискретний випадок.*
     - *1.3.2. Неперервний випадок.*
     - *1.3.3. Але що таке λ?*
   - *1.4. Використання комп’ютерів для автоматизації баєсівського виведення*
     - *1.4.1. Приклад: виведення поведінки на основі даних про обмін текстовими повідомленнями*
     - *1.4.2. Наш перший інструмент: PyMC*
     - *1.4.3. Тлумачення результатів*

1.4.4. *Яка користь можуть надати вибірки з апостеріорного розподілу?*

1.5. *Висновки.*

1.6. *Додаток.*

1.6.1. *Статистичне визначення фактичної розбіжності двох параметрів λ?*

1.6.2. *Узагальнюємо на випадок двох точок розгалуження.*

1.7. *Вправи.*

1.7.1. *Відповіді.*

1.8. *Бібліографія.*

2. *Ще трохи про PyMC.*

2.1. *Вступ.*

2.1.1. *Зв'язки "предок — нащадок".*

2.1.2. *Змінні PyMC.*

2.1.3. *Включення спостережень до моделі.*

2.1.4. *І нарешті…*

2.2. *Підходи до моделювання.*

2.2.1. *Та сама історія, але з іншим кінцем.*

2.2.2. *Приклад: баєсівське A/B-тестування.*

2.2.3. *Простий випадок.*

2.2.4. *A та B разом.*

2.2.5. *Приклад: алгоритм виявлення шахрайства.*

2.2.6. *Біноміальний розподіл.*

2.2.7. *Приклад: шахрайство серед студентів.*

2.2.8. *Альтернативна модель PyMC.*

2.2.9. *Ще декілька хитрощів PyMC.*

2.2.10. *Приклад: катастрофа космічного човна "Challenger".*

2.2.11. *Нормальний розподіл.*

2.2.12. *Що трапилося у день катастрофи "Challenger"?*

2.3. *Чи адекватна наша модель?*

2.3.1. *Розділювальні графіки.*

2.4. *Висновки.*

2.5. *Додаток.*

2.6. *Вправи.*

2.6.1. *Відповіді.*

2.7. *Бібліографія.*

3. *Відкриваємо "чорну скриньку" MCMC.*

3.1. *Баєсівський ландшафт.*

3.1.1. *Вивчаємо ландшафт за допомогою MCMC.*

3.1.2. Алгоритми для MCMC.

3.1.3. Інші наближені методи пошуку апостеріорних розподілів.

3.1.4. Приклад: кластеризація без вчителя з використанням суміші розподілів.

3.1.5. Не мішайте апостеріорні вибірки.

3.1.6. Використання MAP для покращення збіжності.

3.2. Діагностика проблем зі збіжністю.

3.2.1. Автокореляція.

3.2.2. Проріджування.

3.2.3. Функція pymc.Matplot.plot()

3.3. Корисні поради для роботи з MCMC.

3.3.1. Інтелектуальний вибір початкових значень.

3.3.2. Апріорні розподіли.

3.3.3. Народна теорема статистичних розрахунків.

3.4. Висновки.

3.5. Бібліографія.

4. Найвеличніша з несформульованих теорем.

4.1. Вступ.

4.2. Закон великих чисел.

4.2.1. Інтуїція.

4.2.2. Приклад: збіжність пуасонових випадкових змінних.

4.2.3. Як обчислити Var(Z)?

4.2.4. Математичні сподівання та ймовірності.

4.2.5. Який це все має стосунок до баєсівської статистики?

4.3. Некоректна робота з малими числами.

4.3.1. Приклад: агреговані географічні дані.

4.3.2. Приклад: змагання Kaggle (перепис населення США).

4.3.3. Приклад: сортування коментарів на Reddit.

4.3.4. Сортування.

4.3.5. Але для режиму реального часу це надто повільно!

4.3.6. Узагальнення для систем оцінок з привласненням зірок.

4.4. Висновки.

4.5. Додаток.

4.5.1. Диференціювання формули сортування коментарів.

4.6. Вправи.

4.6.1. Відповіді.

4.7. Бібліографія.

5. Що краще: втратити руку чи ногу?

5.1. Функції втрат.

5.1.1. Функції втрат на практиці.

5.1.2. Приклад: оптимізація для раунду “Вітрина” у вікторині “Справедлива ціна”.

5.2. Машинне навчання за допомогою баєсівських методів.

5.2.1. Приклад: передбачення фінансових показників.

5.2.2. Приклад: змагання Kaggle з пошуку темної матерії.

5.2.3. Дані.

5.2.4. Апріорні розподіли.

5.2.5. Навчання та PyMC-реалізація.

5.3. Висновки.

5.4. Бібліографія.

6. Ставимо пріоритети.

6.1. Вступ.

6.2. Суб’єктивні та об’єктивні апріорні розподіли.

6.2.1. Об’єктивні апріорні розподіли.

6.2.2. Суб’єктивні та об’єктивні апріорні розподіли.

6.2.3. Обираємо, обираємо…

6.2.4. Емпіричне баєсівське оцінювання.

6.3. Деякі корисні апріорні розподіли.

6.3.1. Гамма-розподіл.

6.3.2. Розподіл Вішарта.

6.3.3. Бета-розподіл.

6.4. Приклад: баєсівські “багаторукі бандити”.

6.4.1. Застосунки.

6.4.2. Пропонований розв’язок.

6.4.3. Міра якості.

6.4.4. Узагальнення алгоритму.

6.5. Збирання інформації щодо апріорних розподілів у фахівців з предметної області.

6.5.1. Метод рулетки.

6.5.2. Приклад: біржовий прибуток.

6.5.3. Поради професіоналів щодо розподіла Вішарта.

6.6. Спряжені апріорні розподіли.

6.7. Апріорний розподіл Джефріса.

6.8. Вплив апріорних розподілів при збільшенні N.

6.9. Висновки.

6.10. Додаток.

6.10.1. Баєсівський погляд на лінійну регресію зі штрафом.

6.10.2. Вибір виродженого апріорного розподілу.

6.11. Бібліографія.

7. Баєсівське A/B-тестування.

7.1. Вступ.

7.2. Стисле резюме вищенаведеного A/B-тестування конверсій.

7.3. Додаємо лінійну функцію втрат.

7.3.1. Аналіз очікуваного виторгу.

7.3.2. Узагальнення на випадок A/B-експерименту.

7.4. Виходимо за рамки конверсій: t-критерій Стьюдента.

7.4.1. Схема тесту Стьюдента.

7.5. Оцінювання показника зростання.

7.5.1. Створення точкових оцінок.

7.6. Висновки.

7.7. Бібліографія.

6 Самостійна робота студента/аспіранта

Студент витрачатиме 3-4 година на тиждень на самостійну роботу з матеріалом курсу.

Політика та контроль

7 Політика навчальної дисципліни (освітнього компонента)

Студенти отримують бали за правильне та вчасне виконання лабораторних робіт. Загальний рейтинг (кількість балів) складається з: 1) лабораторних робіт (у формі практичних завдань з програмування) 60%, 2) заліку 40%.
Наразі в курсі наявні три лабораторні роботи, кожне оцінюється до 20 балів. Студент повинен здати правильно виконану лабораторну роботу протягом двох тижнів з дня видачі завдання для отримання повної кількості балів, в іншому випадку застосовуються штрафні бали не більше 40% від загальної кількості за лабораторну роботу.

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

Поточний контроль: МКР, лабораторні роботи
Календарний контроль: провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.
Семестровий контроль: залік
Умови допуску до семестрового контролю: зарахування усіх лабораторних робіт, семестровий рейтинг не менше 40 балів.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	*Оцінка*
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

9 Додаткова інформація з дисципліни (освітнього компонента)

- *перелік питань, які виносяться на семестровий контроль (див. додаток до силабусу).*

Робочу програму навчальної дисципліни (силабус):

Складено Професор кафедри інженерії програмного забезпечення в енергетиці, д.е.н., професор А. О. Сігайов

Ухвалено кафедрою інженерії програмного забезпечення в енергетиці (протокол № 28 від 15 травня 2023 р.)

Погоджено Методичною комісією Навчально-наукового інституту атомної і теплової енергетики КПІ ім. Ігоря Сікорського (протокол № 9 від 26 травня 2023 р.)