2025-KELI-002

교과목명
통계적 알고리즘

교수명
이재화

대학명
한림대학교

전공명
데이터사이언스학부
데이터테크전공

AI 활용 수업 사례

KELI

01

AI 활용 수업 계획서

AI 활용 수업 계획서

한림대학교

교과목명	통계적알고리즘		
담당교원	이 재 화	**개설학과 (전공)**	데이터사이언스학부 (데이터테크전공)
개설학기	2025-2	**이수구분**	전공선택
학년	2	**학점**	3
수강인원	26	**수업유형**	이론/실습
활용 AI 도구 (코스웨어 포함)	한림대학교 AI 튜터		

수업개요

본 과목은 다양한 알고리즘의 수학적 원리를 학습하고, R을 활용, 직접 알고리즘을 작성하며 실습해 보는 과목이다. 먼저 선별된 몇 가지 주제에 관하여 배경지식을 학습하고, R을 활용하여 학생 스스로 직접 알고리즘을 작성해 본다. 그리고 AI 튜터를 활용, 짧은 시간에 알고리즘의 핵심 아이디어를 이해하고, 실제 알고리즘을 구현할 때 좋은 가이드로써 도움을 받는다. 이를 통해 졸업 후 실제 데이터를 분석할 때, 적절한 알고리즘을 선택할 수 있는 기초지식을 쌓는 것을 목표로 한다. 또한 AI 튜터의 활용을 통해 AI를 학습에 적절히 활용하는 능력을 키운다.

■ 교과 목표

- 본 과목에서는 다양한 알고리즘의 원리를 학습한다.
- R을 활용하여 직접 알고리즘을 작성하며 실습한다.
- 졸업 후 실제 데이터를 분석할 때, 적절한 알고리즘을 선택할 수 있는 기초지식을 쌓는 것을 목표로 한다.
- AI 튜터의 활용을 통해 AI를 학습에 적절히 활용하는 능력을 키운다.

■ AI 활용의 목적 및 필요성

- 하나의 과목에 여러 가지 주제 및 개념이 동시에 포함되어 있다.
- 학생들이 관련 주제의 배경지식을 익히는 데 어려움이 예상된다.
- AI 튜터의 활용은 짧은 시간에 알고리즘의 핵심 아이디어로 진입하는 데 큰 도움이 될 수 있다.
- AI 튜터는 학생들이 R을 활용하여 알고리즘을 직접 구현해 보는 데 좋은 가이드가 될 수 있다.

■ AI 활용 주차의 수업 운영

- 전반부(2차시)는 관련 주제에 관하여 이론 강의 및 기초 실습을 진행한다.
- 후반부(2차시)는 AI 튜터의 가이드를 받아 제시된 실습 예제를 해결하기 위한 알고리즘을 직접 구현한다.

	■ AI 활용 주차의 학생 활동 ▪ 매주 학습 내용에 관한 유의미한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 자신의 언어로 정리하여 SmartLEAD의 토론 게시판에 공유하는 활동을 한다(최소 1회, 필수). ▪ 차주 수업의 일부는 전주에 학습한 내용 중 핵심 부분을 부연 설명하고, 학생들과 질의/응답 시간으로 활용한다. ▪ 학생들이 AI 튜터에서 제공하는 복습하기를 통해 학습한 개념을 공고히 한다.	
수업목표	**지식 (Knowledge)**	▪ 다양한 알고리즘의 수학적 배경 지식을 이해할 수 있다. ▪ 매주 실습과 과제 및 중간, 기말고사를 통해 학습한 알고리즘의 작동 원리에 대한 이해도를 평가한다.
	기술 (Skill)	▪ R을 활용하여 알고리즘을 직접 구현할 수 있다. ▪ 매주 실습과 과제를 통해 AI 튜터를 적절히 활용할 수 있는 능력을 키운다. ▪ 중간고사, 기말고사를 통해 알고리즘 구현을 위한 R 문법 이해도를 평가한다.
	태도 (Attitude)	▪ AI를 자신의 학습에 적절히 활용하는 것에 대한 긍정적인 태도를 배울 수 있다. ▪ 매주 실습 및 과제를 통해 AI 튜터 활용 능력을 키운다. ▪ 생성형 AI를 포함한 인공지능 기술을 필요에 따라 적절히 수용하는 태도를 갖출 수 있다.

수업계획			
수업운영단계	**주요 내용 및 활동**	**AI 활용 방식**	**평가·산출물**
학습자 사전진단	▪ 개인/집단적 특성: 주로 파이썬 / R 등의 실습을 통한 데이터 분석에 대해 이해하고 있거나 경험이 있는 학습자 ▪ 사전지식 수준/학습 능력: 파이썬 / R 등의 컴퓨터 활용 능력이 좋음, 이론 및 수리적인 지식이 부족함 ▪ AI 활용 능력 및 인식: 챗GPT와 같은 생성형 AI를 활용해 본 경험이 있음 ▪ 학습 성향/학습 동기/태도: 수강생 대부분은 데이터 분석에 대한 이해 또는 경험이 있는 고학년 학습자로 다양한 알고리즘을 이론적으로 이해하기 위한 학습 동기가 높은 편임 ▪ 학습량과 난이도에 따라 학습 속도가 다르나 최신의 AI 기술에 대해서는 많은 관심을 보임	▪ AI 튜터 소개 및 활용 예시 제시 ▪ 핵심 키워드에 관해 질문/답변 해보기 ▪ 퀴즈 생성 등 다양한 조작 해보기	▪ AI 활용 결과 게시판에 공유
AI 활용 방법	▪ 제시된 실습 예제를 해결하기 위한 알고리즘을 직접 구현(AI 튜터의 가이드) ▪ 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드	▪ 수학적 배경 지식에 관한 질의/응답 ▪ R을 활용하여 알고리즘 단계별 구현 가이드	▪ 알고리즘 구현 결과 비교 ▪ 게시판에 업로드

학습지원 및 피드백	▪ 학생 모니터링 및 코딩 풀이, 질문에 대한 피드백 제시 ▪ 토론 게시판에 이해한 바를 자신의 언어로 정리, 공유하도록 유도	▪ AI가 제공하는 복습하기 활동	▪ 보고서에 반영함
사후진단 및 성찰	▪ 학생 보고서를 분석하여 과목 운영에서 미진한 점 보강 ▪ 다양한 배경의 학생 수준을 고려하여 적절한 난이도의 알고리즘에 관한 수업 구성이 필요함 ▪ AI 튜터 개선 사항 반영하여 업데이트 필요함 ▪ 학습 상황별 예시 프롬프트 제공 필요함	▪ 보고서 답변에 대한 요약 ▪ 보고서를 통한 과목 운영 시사점 도출	▪ 학생 보고서 모음

AI윤리방침	▪ 다음과 같이 AI 튜터 활용하여 과제를 제출할 때 본인의 질문 내용과 AI의 답변을 기술하고, 이를 비판적으로 검토하며 본인이 이해한 내용으로 최종 정리 후 구현된 코드와 본인의 코멘트를 함께 기술하도록 하였다. [3주차 과제] ① 수치 미분에 관하여 3주차에 학습하는 동안 AI 튜터에게 질의한 질문과 답변 중 본인에게 도움이 된 것을 본인의 언어로 정리하여 SmartLEAD의 질의응답 게시판에 업로드 하세요. 〈형식〉 *프롬프트 (질문) : 〈-- 본인이 AI 튜터에 입력한 질문 내용을 복사하여 붙여넣기 *AI 튜터의 답변 : 〈-- AI 튜터가 답변한 내용을 복사하여 붙여넣기 *Final Comment : 〈-- AI의 답변을 비판적으로 점검한 후 수정 및 요약, 코드 구현 후 자신의 Comment 추가 ② AI 튜터를 활용하여 도함수와 수치 미분에 관해 스스로 복습하고 그 기록을 제출하세요.
수업 기대효과	▪ AI 튜터를 통해 학생들이 시간과 장소에 구애받지 않고 스스로 학습할 수 있다. ▪ 무한히 질문할 수 있어 학습에서 생기는 궁금증을 적극적으로 질문하며 쉽게 해결할 수 있다. ▪ 필요한 정보들을 빠르게 제공해 주어 학생들이 학습한 개념을 간결하게 요약 정리할 수 있다. ▪ AI 튜터의 답변을 통해 새로운 관점이나 통찰을 얻는 데 도움을 받을 수 있다. ▪ AI 튜터가 제공하는 개념 설명, 퀴즈, 풀이 등을 통해 학습한 내용을 보충할 수 있다.

02

AI 활용 수업 주차별 세부 계획서

AI 활용 수업 주차별 세부 계획서

주차	강의주제	세부내용	수업방식	AI활용방법	과제·산출물
1	오리엔테이션	• 과목 운영 방식 소개 • AI 튜터 결합 수업 소개 • AI 튜터에 익숙해지기	• 강의, 실습	• 본 과목의 핵심 키워드를 AI 튜터를 활용하여 질문하면서, AI 튜터 환경에 익숙해지도록 한다.	• 자기 소개 및 강의에서 기대하는 바를 작성하여 게시판에 업로드
2	R 문법 복습	• R의 기초 문법 복습 • 반복문, 조건문 • 통계, 벡터, 행렬 계산 문제	• 강의, 실습 • AI 튜터의 가이드로 알고리즘 구현	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드	• 본인이 생성한 데이터를 본인이 만든 함수에 적용 후, 내부 명령어와 비교한 결과를 업로드
3	수치 미분	• 도함수, 고계도함수 • 수치 미분 • 차분 근사	• 관련 이론 강의 및 기초 실습 • AI 튜터의 가이드로 알고리즘 구현	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드	• AI 튜터가 제시하는 복습하기 참여
4	수치 적분	• 정적분, 리만합 • 수치적분 • 사다리꼴 공식 • 심프슨 공식	• 관련 이론 강의 및 기초 실습 • AI 튜터의 가이드로 알고리즘 구현	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드	• AI 튜터가 제시하는 복습하기 참여
5	방정식의 해	• 방정식의 해 • 이분법 • 뉴턴의 방법 • 할선법	• 관련 이론 강의 및 기초 실습 • AI 튜터의 가이드로 알고리즘 구현	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드	• AI 튜터가 제시하는 복습하기 참여
6	선형연립방정식	• 선형연립방정식 • 첨가행렬 • 가우스 소거법	• 강의, 실습	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드	• 질문/답변/실습/요약/공유 등 학습활동을 게시판에 업로드
7	복습 1	• 전반기 내용 복습 • 예시 문제 풀이 • 질의 및 응답	• 강의, 실습		
8	중간고사				

9	**선형연립방정식의 반복해법**	• 반복해법 소개 • Jacobi 방법 • Gauss-Seidel 방법	• 관련 이론 강의 및 기초 실습 • AI 튜터의 가이드로 알고리즘 구현	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드 한다.	• AI 튜터가 제시하는 복습하기 참여
10	**켤레기울기법**	• 배경지식 복습 • 양의 정부호, 이차형식 • 켤레기울기법	• 관련 이론 강의 및 기초 실습 • AI 튜터의 가이드로 알고리즘 구현	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드 한다.	• AI 튜터가 제시하는 복습하기 참여
11	**선형회귀**	• 선형회귀 문제 • 정사영, 선형모델 • 정규방정식	• 관련 이론 강의 및 기초 실습 • AI 튜터의 가이드로 알고리즘 구현	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드 한다.	• AI 튜터가 제시하는 복습하기 참여
12	**최적화 알고리즘**	• 경사하강법 • 뉴턴의 방법 • Quasi-Newton	• 관련 이론 강의 및 기초 실습 • AI 튜터의 가이드로 알고리즘 구현	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드 한다.	• AI 튜터가 제시하는 복습하기 참여
13	**행렬의 고윳값**	• 고윳값, 고유벡터 • Power Method • Inverse Power Method	• 관련 이론 강의 및 기초 실습 • AI 튜터의 가이드로 알고리즘 구현	• 학습에 관한 질문과 AI 튜터의 답변을 바탕으로 이해한 바를 본인의 언어로 정리하여 게시판에 업로드 한다.	• AI 튜터가 제시하는 복습하기 참여
14	**복습 2**	• 후반기 내용 복습 • 예시 문제 풀이 • 질의 및 응답	• 강의, 실습		
15	**기말고사**				

03

AI 활용 수업 결과보고서

AI 활용 수업 결과 보고서

수업 운영 결과	
학습자 변화	**교수자 성찰**
■ 수업 전 학습자 상태 ▪ 미적분, 선형대수 배경 없이 수강, 특히 비전공자는 대학수학 수준 자체가 생소함 ▪ Jacobi, Gauss-Seidel, 켤레기울기법 등 처음 접하는 알고리즘과 개념들에 막막함 호소함 ▪ R을 "통계 도구"로만 사용했고, 알고리즘 구현 도구로 쓴 경험 없음 ▪ 공식 암기 위주, 이해보다 결과 도출에 익숙했던 학습 습관을 가지고 있음 ■ 수업 후 학습자 변화 ▪ 알고리즘이 어떤 조건에서, 왜 작동하는지 스스로 묻기 시작함 ▪ 누군가 만들어 놓은 함수의 내부 원리를 인식하여 수렴·오차·계산 비용의 개념을 실제 코드 실행으로 체감 ▪ AI 튜터를 단순 답 요청이 아닌 검증·심화 도구로 활용하는 방식으로 발전 ▪ 밀리지 않고 매주 복습하는 습관 형성 ▪ 어려운 문제 앞에서 이해될 때까지 반복하는 끈기 형성 ▪ 다양한 알고리즘(뉴턴법 vs 할선법, Jacobi vs Gauss-Seidel 등)의 장단점 비교·선택 가능 ▪ R로 알고리즘을 바닥부터 구현하고 내장 함수 결과와 비교 검증하는 능력 형성 ▪ 낯선 문제에도 수학적 원리를 파악하고 알고리즘적으로 접근하는 사고 틀 획득 ■ 대표 학습자 반응 ▪ “수식이 R 코드로 어떻게 변환되는지 질문하고 토론했다. ‘왜 이 알고리즘을 쓰는가?’에 집중하여 직접 코드를 작성하고 내장 함수 결과와 비교하는 ‘검증 중심’ 학습을 했다.” ▪ “퍼즐 조각들이 깔끔하게 맞아 떨어지듯이 개념들이 쓰이는 방식에 즐거움을 느꼈다.” ▪ “처음엔 코드 만드는 것이 정말 어려웠지만, 알고리즘이 어떤 식으로 작동하는지, 어떤 때에 쓰는지 알 수 있는 수업이었다.”	(1) 수업 전 계획 ■ AI 결합 전 ▪ 알고리즘마다 필요한 다양한 배경지식을 강의하는 데에 시간을 많이 할애하므로 다루고자 하는 주제에 제한이 있었으며, 실습을 진행할 시간도 부족하였다. ▪ 학생들의 R에 대한 사전 지식이나 경험 정도가 다양하여 실제 알고리즘을 구현해 보는 것에도 제약이 많았다. ■ AI 결합 후 ▪ 한 학기 강의 주차 내에 다룰 수 있는 알고리즘의 주제를 늘릴 수 있었다. ▪ AI 튜터의 가이드를 받아 알고리즘 구현을 진행하므로 간단한 주제에서부터 심화된 주제까지 다양한 난이도를 경험하게 할 수 있었다. (2) 수업 중 운영 ■ AI 결합 전 ▪ 학생들의 다양한 수준의 배경지식으로 인해 수업/질의응답/코딩실습 과정에서 개인화된 교육이 어려웠다. ■ AI 결합 후 ▪ AI 튜터를 통해 동일한 시간을 사용하고도 알고리즘의 핵심 아이디어에 접근하는 데 도움이 되었다. ▪ 학습에서 생기는 궁금증을 적극적으로 질문하며 쉽게 개념을 이해할 수 있었다. ▪ AI 튜터의 가이드를 통해 R에 대한 지식이 부족하여도 알고리즘을 구현해 볼 수 있었으며 시간이 지남에 따라 점점 익숙해지게 되었다. (3) 수업 후 평가 ■ AI 결합 전 ▪ 매주 학습 상황을 점검하는 다양한 과제를 제시하느라 많은 시간을 할애하였다.

▪ “처음부터 완벽히 이해해야 한다는 부담감을 줄일 수 있었고, 오류가 났을 때 두려워하기보다 어떤 부분인지 살펴보는 과정에 익숙해졌다.” ▪ “이해하지 못한 원리가 많아 아쉬움이 남는다. 한편으론 필요할 때 필요한 부분을 알아내고 적용하는 법을 배운 게 이번 수업의 결론이 아닌가 싶다.” ▪ “주성분 분석과 요인분석의 가정이 반대 방향이라는 점, 같은 분석이라도 가정이 다르면 완전히 다른 해석이 나온다는 것을 체감했다.”	■ AI 결합 후 ▪ 시간과 공간에 관계없이 궁금한 것을 AI 튜터에게 질문할 수 있으므로 쉽게 개념을 복습하고 이해할 수 있었다. ▪ AI 튜터가 제시하는 복습하기를 통해 AI의 질문에 답하며 학습한 내용을 보충할 수 있었다. ▪ 학생의 학습 진행 상황(과제, 질의응답, 복습하기 등)을 보고서 양식에 모두 모아 전체적으로 평가할 수 있었다.

우수 과제 사례

▪ 선형회귀와 최소제곱문제 단원의 개념 이해 과정에서 AI 튜터를 활용한 사례

최소제곱문제 (least squares problem)

$$\| A\bar{u} - \mathbf{y} \| = \min \| A\mathbf{u} - \mathbf{y} \|$$

- 최소제곱문제는 결국 y와 Au사이의 거리(distance)를 가장 짧게 만드는 $\bar{u}$를 찾는 것
- 즉 각 데이터 (x_i, y_i)에 대하여, $\hat{y}_i = a + bx_i$라 하면 (제곱) 오차 $(y_i - \hat{y}_i)^2$의 합이 최소가 되는 a, b를 구하는 것.

질문 : 선형회귀에 대해서 자세하게 설명해줘.

선형 회귀는 주어진 데이터에 대해 두 변수 간의 관계를 선형적으로 모델링하는 기법입니다. 이는 주로 하나의 종속 변수(반응 변수)와 하나 이상의 독립 변수(설명 변수) 간의 관계를 설명하는 데 사용됩니다. 선형 회귀의 목표는 종속 변수의 값을 예측하기 위해 독립 변수의 가중치를 결정하는 것입니다.

선형 회귀의 기본 형태는 다음과 같습니다:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \epsilon$$

여기에서 y는 종속 변수, x_1, x_2, ⋯, x_n 는 독립 변수, β_0는 절편, β_1, β_2, ⋯, β_n는 각 독립 변수의 계수, ϵ은 오차(term)입니다.

선형 회귀의 주요 목표는 데이터에 가장 잘 맞는 직선을 찾는 것입니다. 이를 위해 최소제곱법(Least Squares Method)을 사용하여 각 데이터 포인트와 회귀선 사이의 오차 제곱합을 최소화합니다. 오차는 실제 값과 예측 값 간의 차이이며, 이를 줄임으로써 모델의 정확도를 높입니다. 추가적으로, 정규방정식을 사용하여 회귀계수를 직접 계산할 수 있지만, 이는 오차에 민감하므로 대규모 데이터나 다차원 데이터에서는 QR 분해와 같은 방법을 사용하여 안정적으로 해를 찾습니다.

통계적알고리즘 수업에서는 이러한 개념을 이해하고 실제로 프로그램을 통해 구현하는 것을 목표로 합니다. 이를 통해 수학적 이론뿐만 아니라, 실질적인 데이터 분석 기술을 습득할 수 있습니다.

[출처] 9주차 W9-1차시-강의자료파일, 11주차 W11-1차시-강의자료파일

질문 : 선형회귀와 비선형회귀를 나누는 기준이 뭐야?

선형 회귀와 비선형 회귀를 나누는 기준은 모델의 수학적 관계에 있습니다.

선형 회귀:

• 모델이 선형 방정식 형태를 띄는 경우입니다.

• 종속 변수 y와 독립 변수 x 사이의 관계가 선형적입니다. 즉, 데이터 포인트가 직선으로 표현될 수 있습니다.

• 일반적인 형태는 다음과 같습니다: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \epsilon$

• 여기서 ϵ은 오차를 나타내며, β_1, β_2, ⋯, β_n는 회귀 계수입니다.

• 최소제곱법을 사용하여 회귀선을 맞추며, 주로 정규방정식이나 QR분해를 통해 해를 찾습니다.

비선형 회귀:

• 종속 변수와 독립 변수 사이의 관계가 비선형적입니다. 즉, 데이터 포인트가 곡선으로 표현될 수 있습니다.

• 모델이 비선형 함수로 표현됩니다. 예를 들어, 다항식 회귀, 지수 회귀, 로그 회귀 등이 있습니다.

• 비선형 회귀는 선형 회귀보다 복잡하며, 해를 찾기 위해 최적화 알고리즘이 필요합니다.

통계적알고리즘 수업에서는 이러한 개념을 이해하고 실제로 프로그램을 통해 구현하는 것을 목표로 합니다. 이를 통해 수학적 이론뿐만 아니라, 실질적인 데이터 분석 기술을 습득할 수 있습니다.

[출처] 11주차 W11-1차시-강의자료파일, 10주차 W10-2차시-강의자료파일

질문 : 최소제곱법을 이용한 다항식 곡선은 비선형회귀야?

최소제곱법을 이용한 다항식 곡선은 비선형 회귀에 해당합니다. 비선형 회귀는 데이터가 비선형적인 관계를 가질 때 사용되는 모델링 기법입니다. 다항식 회귀는 독립 변수의 다항식 형태로 종속 변수를 설명하는 방식으로, 종속 변수와 독립 변수 사이의 관계가 직선이 아닌 곡선으로 표현됩니다.

다항식 회귀는 선형 회귀와 달리 방정식의 형태가 비선형이기 때문에, 다항식의 차수가 1보다 큰 경우에는 비선형 회귀로 간주됩니다. 이러한 모델을 적합하기 위해서는 데이터와 곡선 사이의 오차 제곱합을 최소화하는 방법이 사용되며, 이는 최소제곱법을 통해 수행됩니다. 통계적알고리즘 수업에서는 이러한 개념을 이해하고 실제로 구현하며, 비선형 회귀를 포함한 다양한 회귀 기법을 학습합니다.

[출처] AI 기반 응답입니다.

Gemini 질문 : 최소제곱곡선은 선형회귀야?

결론부터 말씀드리면, 최소제곱법을 이용해 다항식 곡선(예: $y = a + bx + cx^2$)을 구하는 것도 "선형 회귀(Linear Regression)"의 범주에 포함됩니다. 직관적으로는 그래프가 "곡선"이라서 비선형(Non-linear) 같아 보이지만, 수학적으로는 선형으로 취급합니다. 그 이유는 다음과 같습니다.

1. "선형"의 기준은 계수(a, b, c)입니다.

회귀 분석에서 '선형'이라는 말은 x(변수)가 직선이냐 아니냐가 아니라, 우리가 구해야 할 계수들(a, b, c)이 선형 결합으로 되어 있느냐를 의미합니다.

선형 회귀 (O): $y = a + bx + cx^2$

여기서 x와 x^2은 우리가 이미 알고 있는 데이터(숫자)입니다.

우리가 모르는 미지수 a, b, c 입장에서는 모두 1차식이므로 선형 문제입니다.

비선형 회귀 (X): $y = a\sin(bx)$ 또는 $y = e^{ax}$

이 경우 미지수 b나 a가 삼각함수나 지수함수 안에 갇혀 있어서, 단순한 행렬 풀이로는 값을 구할 수 없습니다. 이것이 진짜 비선형입니다.

	2. 수업 자료 근거 11주차 강의 자료를 보면 이 점이 명확히 드러납니다. 1차 직선 찾기: $y=a+bx$를 찾을 때 행렬을 이용해서 $A\mathbf{u}=\mathbf{y}$꼴로 만들고 풀었습니다. 2차 곡선 찾기 (Curve Fitting): $y=a+bx+cx^2$을 찾을 때도, x^2을 하나의 새로운 변수처럼 취급하여 똑같은 형태의 선형연립방정식($A\mathbf{u}=\mathbf{y}$)을 만들어서 풉니다. 즉, x를 제곱하든 세제곱하든, 계수(a, b, c) 입장에서 식이 1차 결합이라면, 그것은 "다중 선형 회귀(Multiple Linear Regression)" 모델이 됩니다. 따라서 최소제곱곡선을 구하는 과정도 선형 회귀 분석의 응용입니다. **요약** 선형회귀 : 주어진 데이터에 대해 두변수 간의 관계를 선형적으로 모델링하는 기법. 최소제곱법 : $\mathbf{y}$와 $A\mathbf{u}$사이의 거리를 가장 짧게 만드는 $\mathbf{u}$를 찾는 것. 각 데이터들 (x,y)에 대해서 $y=a+bx$라 하면 제곱오차의 합이 최소가 되는 a, b를 구하는 것. 정규방정식을 사용하여 회귀계수를 직접 계산할 수 있지만 이는 오차에 민감하여 대규모 데이터나 다차원 데이터에서는 QR분해와 같은 방법을 사용하여 안정적으로 해를 구함. **질문** : 선형회귀(linear regression)는 일차함수로 표현이 되기 때문에 최소제곱법을 이용한 다항식 곡선은 선형회귀가 아니라고 생각되는데, AI튜터와 Gemini의 대답에 헷갈립니다. 답변 : 데이터를 잘 나타내는 다항식 곡선을 구하는 것을 curve fitting이라 하는데, 이를 계산하기 위해서 최소제곱법을 이용하는 것은 (원리적으로) 선형적인 방법에 해당한다. **핵심 보완** : 선형회귀 vs 비선형회귀 Q. $y=ax^2+bx+c$(곡선)를 구하는데 왜 '선형' 회귀인가? **핵심 기준 :**'선형'의 기준은 변수 x의 모양이 아니라, 우리가 구해야 할 계수(파라미터 a, b, c)가 선형 결합인가 이다. **판별법** : 선형 회귀 : 계수들이 덧셈으로 연결됨. $y=a+bx$ (직선) -> 선형 $y=a+bx+cx^2$ (곡선) -> x^2을 x_2라는 새로운 변수로 보면 $y=a+bx_1+cx_2$ 꼴이므로 선형이다. (행렬 $A\mathbf{u}=\mathbf{y}$로 풀 수 있다.) 비선형 회귀 : 계수의 지수, 로그, 삼각함수 안에 갇혀 있거나 서로 곱해짐 $y=e^{ax}+b$ -> 계수 a가 지수 안에 있음. 행렬로 한 번에 풀 수 없고 최적화 알고리즘이 필요.
교육적 성과	■ AI 활용 수업에 대한 교수자의 소감 요약 ▪ AI 튜터는 기본적으로 강의자료를 중심으로 답을 제시해 주므로 학생들의 AI 활용 시 발생할 수 있는 할루시네이션 문제에 관한 걱정을 덜 수 있었다. ▪ 제한된 한 학기의 강의 시간 안에서 이전보다 더 다양한 학습 주제를 선정하여 다룰 수 있었다. ▪ 강의자료 준비 시 AI 활용으로 교수자 부담을 완화할 수 있었다. ■ AI 활용 수업에 대한 학습자의 소감 요약 ▪ AI 튜터를 이용한 학습은 단순 검색이 아닌 같이 학습할 수 있는 도구였다. ▪ AI 튜터를 통해서 궁금했던 점들을 질문해 보고 답변을 들으면서 잘 못 알고 있던 내용을 수정해 볼 수 있었다.

	▪ 여러 방법의 코드들을 물어보면서 스스로 해봤던 코드들과 비교해 볼 수 있었다. ▪ 매 주차 복습해 볼 수 있어서 하나의 개념을 완벽하게 이해하고 계속해서 생각해 볼 수 있었다. ▪ 수업에서 이해하지 못했거나 필기를 못한 부분을 AI 튜터를 이용해서 복습을 진행하였다. 특히 예시나 단계별 설명을 통해 알고리즘의 흐름을 이해하는 데 유용했다. ▪ 강의자료를 기반으로 개념 설명과 문제 풀이 방향을 제시해 준 점은 학습에 도움이 되었다. 특히 수식이나 알고리즘의 흐름을 정리할 때, 강의 내용을 다시 구조화해 이해할 수 있었다. ▪ 시간과 공간의 제약 없이 누구나 편리하게 사용할 수 있었다. ▪ 언제든지 궁금한 게 생기면 바로 물어보고 즉각적인 답변을 얻을 수 있었다. ▪ AI 튜터는 짧은 시간 안에 답변을 해주어 그 자리에서 바로 코드 문제를 해결할 수 있었다. ▪ 개념 질문 시 정리를 해주어 시험공부할 때 개념 정리에도 도움이 많이 되었다.
한계 및 개선사항	■ 한계점 ▪ 다양한 형식의 퀴즈 출제가 필요하다. ▪ 현재의 AI 튜터 활용 시 수학 과목 특성상 원하는 형태의 퀴즈(예제와 유사한 유형의 수학 문제 등)가 출제되지 않고 개념을 묻는 문제만 반복하여 출제되었다. ▪ 본래 수업에서 계획했던 AI 튜터가 제시하는 퀴즈는 진행하지 못하고 '복습하기'로 대체하였다. ■ AI 튜터에 대한 학생 의견 ▪ 하나 질문 후 답변을 듣고 그 답변에 대해 또 연속되는 질문을 하게 되면 답변이 이어지지 않고 다른 답변을 하게 되어 답답한 부분이 있었다. 기존의 질문들을 저장하고 있어 물어봐도 계속 대답을 해줄 수 있게 개선되었으면 한다. ▪ 질문의 의도와 다르게 너무 교과서적인 답변만 반복하는 경우가 있어, 문맥을 더 잘 파악하여 맞춤형 예시를 들어주는 기능이 강화되면 좋겠다. ▪ 강의자료 내 페이지나 슬라이드 번호를 함께 제시해 준다면 학습 효과가 더욱 높아질 것으로 생각한다. ▪ AI 튜터의 답변을 복사하여 스마트 리드에 붙여 넣으면 수식이 제대로 표시되지 않거나, R에서 바로 사용할 수 있는 함수 형태로 변환되지 않아 불편함이 있었다.
시사점	■ 수업 설계 ▪ 이론-실습 연계 구조의 유효성 확인 : 수학적 배경지식 → R코드 구현 → 결과 검증으로 이어지는 3단계 학습 흐름이 다수 학생에게 효과적으로 작동함. 단순 공식 암기가 아닌 알고리즘 원리 이해 중심 수업 설계가 학습 동기 및 이해도 향상에 기여 ▪ 난이도 진입 장벽 관리 필요 : 수학적 배경(미적분, 선형대수)이 부족한 학생들이 초반 탈락 위기를 경험함. 수업 초반부에 수학적 선수 지식을 보완하는 브릿지 콘텐츠 또는 별도 보충 자료 제공 필요함. ▪ 알고리즘 비교·선택 관점의 명시적 교수 설계가 필요 : 각 알고리즘의 수렴 조건·계산 비용·적용 조건을 비교표 형태로 제공하는 설계 보완 필요 ▪ 질의응답 활동의 복습 효과 입증 : 질의응답 게시판이 단순 질문 공간을 넘어 동료 학습 플랫폼으로 기능함. 매주 학습 내용을 게시판에 정리·공유하는 구조적 복습 과제가 장기 기억 및 이해 심화에 효과적임. 과제 마감 전 한꺼번에 몰아서 수행하는 것의 방지를 위해 주차별 최소 참여 기준 설정이 필요함 ▪ 전공 배경에 따라 다른 학습 경로를 제공하는 차별화된 학습 설계 또는 학습 지원 체계 마련 필요할 수 있음.

	■ AI 튜터 활용 ▪ 즉각적 피드백 기능의 학습 효과 : 수업 중 실시간으로 막히는 개념과 R 코드 오류를 즉시 해결하는 도구로 활용 → 학습 흐름 단절 최소화. 교수자 질문 기회가 제한된 대규모 강의 환경에서 개인 맞춤 질의 창구로서 높은 유용성 확인 ▪ 복습 과제 연계 방식의 효과성 검증 : 매주 AI 튜터를 활용한 복습 과제 수행이 학습 내용 장기 기억에 기여. 자발적 복습 동기가 낮은 학생도 과제 형식 덕분에 반강제적 복습 수행 → 구조화된 AI 과제 설계의 필요성 입증 ▪ 대화 맥락 연속성 부재가 가장 큰 개선 과제 : 다수 학생이 공통적으로 지적했듯이 연속 질문 시 이전 답변과 맥락이 끊겨 반복 설명이 필요함. ▪ 답변 정밀도 및 의도 파악 능력 개선 필요 : 질문 의도와 다른 답변 제공, 교과서적 반복 답변, 지나치게 복잡한 R 코드 제시 등의 한계 보고. 학습자의 이해 수준을 자동 감지하고 난이도를 조절하는 적응형 응답 기능 개발 필요함. 수업에서 실제 사용한 코드 스타일과 일치하는 답변 제공을 위한 강의자료 기반 파인튜닝 검토 필요함. ■ 대학 확산 측면 시사점 ▪ AI 튜터의 교과 연계형 도입 모델로서 검증 가능성 확인 : 특정 강좌의 강의자료와 연동 된 AI 튜터 운영이 학생 만족도 및 학습 효과 면에서 긍정적 반응 확인. 범용 AI(ChatGPT 등)와 차별화된 교과 특화형 AI 튜터의 대학 도입 모델로서 확산 가능성 있음 ▪ 교수자 부담 경감 및 학습지원 확장 수단으로 활용 가능 : 학생 개별 질문을 AI가 1차 처리 → 교수자는 심화 개념 및 오개념 교정에 집중 가능. 대규모 강의, 비대면 강의 환경에서의 학습 격차 완화 도구로 확산 가능성 높음 ▪ 학습 데이터 축적을 통한 교육 개선 가능성 : AI 튜터 활용 이력(질문 주제, 오류 유형, 반복 질문 패턴 등)이 교수자의 수업 개선 피드백 데이터로 활용 가능. 학습분석 기반 교육 개선 체계 구축의 인프라로 기능 가능 ▪ 학습자 자기 주도 역량 강화 경험의 확산 가치 : 다수의 학생이 "스스로 문제를 찾고 해결하는 태도"를 이 수업을 통해 처음 경험했다고 보고. 단순 지식 전달을 넘어 자기주도 학습 역량 함양 과목으로의 재설계 가능성 → 교양·전공 모두 적용 가능한 고등교육 모델로서 확산 가치 높음

시각자료

■ AI 튜터 접속 화면(https://tutor.k-university.ai/)





AI 튜터

4주차 학습 내용을 요약해 드리겠습니다.

1.

알고리즘의 수학적 아이디어와 배경 지식

- **심프슨 공식(Simpson's Rule)**: 심프슨 공식은 **정적분을 근사적으로 계산**하기 위한 방법입니다. 구간 $[a, b]$를 짝수 개수의 등분점으로 나누고, 각 구간에서 함수 $f(x)$를 **이차 다항식으로 근사**하여 적분 값을 계산합니다. 그 결과, 더욱 정밀한 근삿값을 얻을 수 있습니다.

2.

간단한 예시 문제

- **문제 설명**: 다음의 정적분을 심프슨 공식을 사용하여 근사하세요.

$$\int_1^3 \sqrt{4-x}\,dx$$

- **해결 과정**:

1. 구간 $[1, 3]$을 4등분하여 $x_0 = 1, x_1, x_2, x_3 = 3$로 나눕니다.
2. 각 $y_k = f(x_k)$를 계산합니다.
3. 심프슨 공식을 적용하여 근삿값을 구합니다:

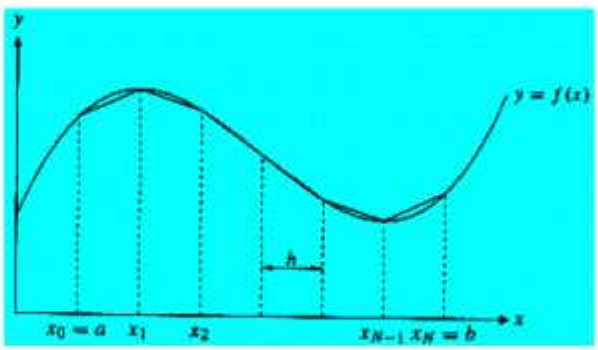
$$\int_1^3 \sqrt{4-x}\,dx \approx \frac{h}{3}\left(y_0 + 4y_1 + 2y_2 + 4y_3 + y_4\right)$$

■ AI 튜터 활용 예시

(1) 학생들이 AI 튜터와의 질의 응답 과정을 바탕으로 본인의 언어로 정리한다.

- 활용한 AI 서비스: AI 튜터
- AI 활용 주체: 학습자
- AI 활용 목적 및 방법: 제시된 실습 예제를 해결하기 위해 학습한 알고리즘을 구현해야 한다.
- AI 튜터를 활용하여 알고리즘 구성에 대한 가이드를 받는다.

$$\int_a^b f(x)dx \approx \frac{h}{2}\{(y_0+y_n)+2(y_1+y_2+\cdots+y_{n-1})\}$$



[실습] $n=8$일 때 사다리꼴 공식을 사용하여 정적분 $\int_0^2 \sin x^2 dx$의 근삿값을 구하여라.

[4주차 과제] ① 수치 적분에 관하여 4주차에 학습하는 동안 AI 튜터에게 질의한 질문과 답변 중 본인에게 도움이 된 것을 본인의 언어로 정리하여 SmartLEAD의 질의응답 게시판에 업로드 하세요.

〈형식〉
*프롬프트 (질문) : 〈-- 본인이 AI 튜터에 입력한 질문 내용을 복사하여 붙여넣기
*AI 튜터의 답변 : 〈-- AI 튜터가 답변한 내용을 복사하여 붙여넣기
*Final Comment : 〈-- AI의 답변을 비판적으로 점검한 후 수정 및 요약, 코드 구현 후 자신의 Comment 추가

② AI 튜터를 활용하여 정적분과 수치 적분에 관해 스스로 복습하고 그 기록을 제출하세요.

Q1. 수치적분에 대해 설명해줘

A1. **수치 적분**(Numerical Integration)은 함수의 정적분 값을 근사적으로 계산하는 방법입니다. 수치 적분이 필요한 이유는 모든 함수의 부정적분을 초등함수 형태로 표현할 수 있는 것은 아니기 때문입니다. 따라서, 정적분의 값을 직접 계산하기 어려운 경우 근삿값을 사용하게 됩니다.

수치 적분의 기본 아이디어는 함수의 그래프 아래 면적을 작은 구간들로 나누고, 각 구간의 면적을 합산하여 전체 면적을 근사하는 것입니다. 이 방법은 리만 합의 극한을 사용하며, 구간을 충분히 작게 나누면 리만 합의 극한으로 정적분의 근삿값을 얻을 수 있습니다.

대표적인 수치 적분 기법으로는 **사다리꼴 공식**과 **심프슨 공식**이 있습니다.

- **사다리꼴 공식**은 구간을 여러 개의 작은 사다리꼴로 나누어 면적을 계산하는 방법입니다.
- **심프슨 공식**은 포물선을 사용하여 구간을 근사적으로 표현하는 방식으로, 보다 정밀한 결과를 제공합니다.

이러한 기법들은 수치 해석에서 중요한 역할을 하며, 복잡한 함수의 적분을 효율적으로 계산하는 데 사용됩니다. **중점 법칙**과 같은 다른 리만 합 변형 또한 존재하며, 특정 상황에서 더 나은 근삿값을 제공할 수 있습니다. [출처] 4주차 W4-2차시-강의자료파일

Q2. 수치적분의 내용 중에서 사다리꼴 공식에 대해 설명해줘. R에서 사용할 수 있는 코드와 함께 예시를 들어 설명해줬으면 좋겠어

A2. 사다리꼴 공식은 수치 적분에서 널리 사용되는 기법 중 하나로, 주어진 구간을 여러 개의 작은 사다리꼴로 나누어 각 사다리꼴의 면적을 합산하여 정적분의 근삿값을 계산하는 방법입니다. 이 공식은 구간 [a,b]를 n등분하여 각 구간의 끝점을 연결하여 사다리꼴을 형성합니다. 이때 각 소구간의 길이를 h=(b−a)/n라고 하고, 주어진 함수 f(x)의 값을 yk=f(xk)로 표현합니다. 사다리꼴 공식은 다음과 같이 수식으로 표현됩니다:

$$\int_a^b f(x)\,dx \approx h2(y0+yn+2(y1+y2+\cdots+yn-1))$$

R 코드 예시:

```
trapezoidal_rule <- function(f, a, b, n) {
  h <- (b - a) / n
  x <- seq(a, b, by = h)
  y <- f(x)
  integral_approx <- h * (sum(y) - (y[1] + y[n+1]) / 2)
  return(integral_approx)
}
# 예시 함수
f <- function(x) { sin(x^2) }

# 적분 구간과 분할 수 설정
a <- 0
```

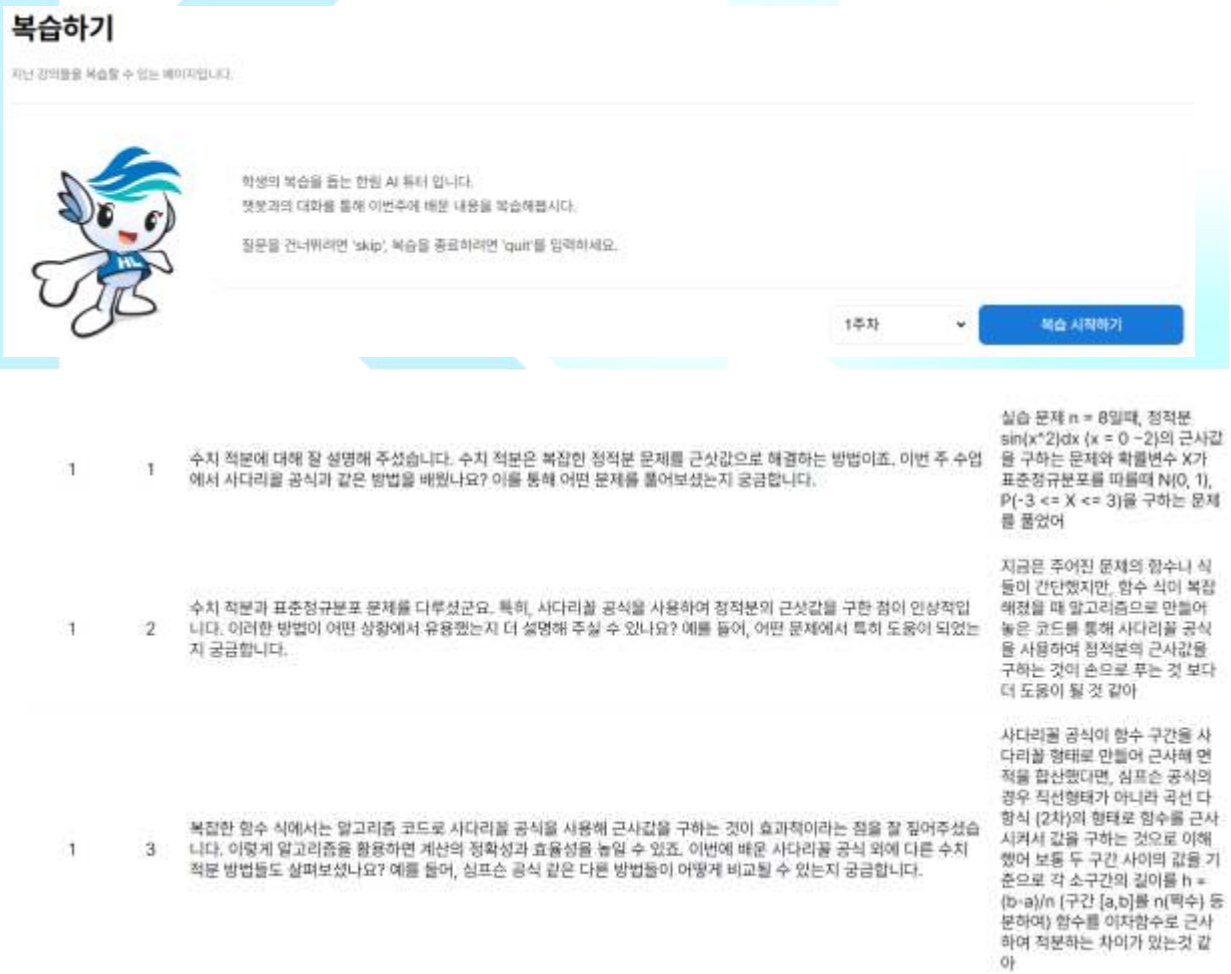
```
b <- 2
n <- 8

# 사다리꼴 공식을 통한 근삿값 계산
result <- trapezoidal_rule(f, a, b, n)
print(paste("근삿값:", result))
```

이 코드는 함수를 입력받아 사다리꼴 공식을 사용하여 [a,b] 구간에서의 적분 근삿값을 계산합니다. trapezoidal_rule 함수는 입력 함수 f, 구간의 시작점 a, 끝점 b, 그리고 분할 수 n을 받아 적분값을 근사합니다. 이 예시에서는 f(x)=sin(x2) 함수에 대해 구간 [0,2]에서 사다리꼴 공식을 적용하여 적분 값을 계산합니다.
[출처] 4주차 W4-2차시-강의자료파일

#comments
사다리꼴 공식이 잘 이해가 가지 않았기에 사다리꼴 공식에 대한 내용을 위주로 튜터에게 질문을 하였다. 사다리꼴 공식을 R에서 사용할 때, 분할 수 선택, 함수의 연속성, 함수의 형태, 계산 오차, 함수의 안정성 부분에 대해 주의하여 사용하여야 원하는 근사값을 효율적으로 얻을 수 있다는 걸 알게 되었다.

(2) 매주 AI 튜터를 활용하여 수행한 복습하기 활동을 제출한다.



■ 학습 기록을 담은 보고서
- 한 학기 동안 학생의 학습활동을 보고서로 제출한다.
- 개인 성찰 노트, 참여 부분, 최종 소감으로 구성되어 있다.

▪ 학생 보고서 모음(일부)은 본 보고서의 [별첨]을 참고

[보고서 양식]

PBL Report (개인 성찰 노트)

Fall, 2025

PBL Report (Form)

Your Class: 통계적알고리즘

Lecturer : 이 재 화

과제 Due day : 2025년 12월 14일

Name (이름) :
Student # (학번) :
e-mail (이메일):

PBL Report (개인 성찰 노트)

Fall, 2025

PBL Report (Form)

Your Class: 통계적알고리즘

Lecturer : 이 재 화

과제 Due day : 2025년 12월 14일

Name (이름) : ***
Student # (학번) : ********
e-mail (이메일): *******

1장. 개인 성찰 노트

[다음 질문에 답변하세요.]

1. 나는 지금 수행되고 있는 학습의 진행 내용을 이해하고 있는가?

통계적 알고리즘 수업을 들으며 다양한 알고리즘들에 대해 학습할 수 있었습니다. 먼저, 알고리즘을 학습하기 전 기초가 되는 수학적 배경과 통계적 지식을 이해하였습니다. 도함수, 고계 도함수, 미분과 같은 핵심 미분 개념을 이해한 뒤, 차분 근사에 대한 원리를 이해할 수 있었습니다. 이후 이러한 배경 지식을 바탕으로 사다리꼴 공식, 심프슨 공식, 이분법, 뉴턴의 방법, 선형 연립방정식의 반복해법, 콜레스키 기울기법을 공식에 직접 적용하며 이해를 확장할 수 있었습니다. 이러한 과정을 통해 최적화 알고리즘 경사하강법, 뉴턴의 방법, Quasi-Newton, Power Method 등을 이해할 수 있었고 다양한 알고리즘에서 기초가 되는 수학적인 배경과 지식을 학습한 뒤 이를 통해 사용하고자 하는 알고리즘을 단계적으로 이해하며 학습할 수 있었습니다.

네. 이해하고 있습니다. 통계적 알고리즘 강좌는 수학 공식을 암기하고 풀이하는 것이 아니라, 미분, 적분, 방정식을 배우고, 선형 대수(연립방정식, 행렬)를 이해하고 최적화 및 데이터 분석에 적용하는 방식으로 수업이 진행되었습니다. 알고리즘 원리를 학습하고, R을 활용하여 직접 알고리즘을 작성하여 실습 해보는 과목이었습니다. 이를 통해서 수강 후 실제 데이터를 분석할 때, 적절한 알고리즘을 선택할 수 있는 기초 지식을 쌓는 것을 목표로 수업에 임했습니다. 특히 컴퓨터가 수학적 문제를 해결하기 위해서 '반복법'을 사용해 해를 찾아가는 과정과 원리를 확실하게 이해할 수 있었습니다.

강의를 처음 들을 때, 수업에서 다루는 개념과 알고리즘의 구체적인 흐름을 완전히 이해하기는 어려웠습니다. 생전 처음 듣는 알고리즘의 종류가 많았기에 알고리즘의 단계별 의미를 파악하는 것뿐만 아니라, 각 단계가 전체 문제 해결 과정에서 어떤 역할을 하는지 연결지어 생각하는 것이 매우 어려웠습니다. 특히 여러 알고리즘이 서로 다른 상황에서 어떻게 활용되는지, 차이점이나 주의해야 할 점과 함께 고려해야 할 점이 많아 더 어렵고 복잡하게 느껴졌습니다. 개념을 읽거나 강의를 듣는 것만으로는 충분히 이해되지 않아, 혼자 고민하는 시간이 길어지기도 했습니다. 그러나 저는 이러한 어려움을 외면하지 않고, 학습 내용을 정리하고 반복적으로 검토하는 방식을 취했습니다. 강의안과 관련 자료를 여러 차례 확인하며 개념 간의 관계와 알고리즘의 흐름을 구조화하였고, 이를 통해 학습 내용을 점검하고 이해도를 확인했습니다. 처음에는 막연하게 느껴졌던 개념들이 조금씩 명확해지면서, 각 알고리즘이 문제 해결 과정에서 어떤 역할을 수행하는지 이해할 수 있게 되었습니다. 또한, 학습 과정에서 자신이 이해하지 못한 부분이 무엇인지 스스로 확인할 수 있었고, 그 부분을 위주로 살펴보며 완전히 이해하고자 하였습니다. 지금은 학습 내용을 단순히 암기하거나 표면적으로 이해하는 수준을 넘어, 개념과 알고리즘의 흐름을 체계적으로 이해하고, 스스로 문제를 분석하며 해결할 수 있는 능력을 갖추게 되었습니다. 필요할 경우 다른 사람에게 해당 알고리즘과 개념을 설명할 수 있을 정도로 이해가 깊어졌으며, 처음에는 어렵게만 느껴졌던 내용이 이제는 어느정도 정리되어 두렵지 않게 되었습니다. 이러한 학습 과정을 통해, 저는 단순히 강의를 듣고 이해하는 수준을 넘어, 익숙한 문제가 아니더라도 어느정도 적용할 수 있을만큼 강의 내용을 이해하고 있습니다.

2. 어떤 방법을 통해서 학습하였는가? (학습 방법 및 자료)

이전 학기에 수강한 [데이터 구조와 컴퓨팅]과 [행렬과 데이터 분석] 강의와 내용이 겹치는 부분이 많아, 이 부분은 비교적 어렵지 않게 이해할 수 있었습니다. 그러나 알고리즘을 구상하는 방법과 같은 새

로운 개념은 쉽게 이해가 되지 않아, 이 부분을 깊이 있게 이해하기 위해 많은 시간을 사용했습니다. 먼저, 그 주차에 해당하는 강의안을 여러 차례 반복해서 살펴보며 알고리즘이 어떤 흐름으로 구성되는지를 이해하고자 노력했습니다. 단순히 개념을 읽는 데 그치지 않고, 알고리즘의 각 단계가 왜 필요한지, 입력과 출력, 그리고 내부 연산이 어떤 관계를 가지는지를 확인했습니다. 이후, 교수님이 제공하신 코드나 수업 시간에 작성한 코드를 반복해서 직접 작성하며 실습하였습니다. 이 과정에서 코드 일부가 이해되지 않거나, 출력 결과를 이해할 수 없어 혼란스러운 경우도 많았습니다. 이때는 챗 GPT나 AI 튜터를 활용해 모르는 부분을 질문하고, 이해가 될 때까지 상세히 분석하며 반복 학습을 이어갔습니다. 단순히 코드를 따라 치는 수준에서 벗어나, 스스로 문제를 해결하고, 다른 친구에게 알고리즘과 코드를 설명할 수 있을 정도가 되었을 때, 개념과 코드에 대한 완전한 이해가 이루어졌다고 판단했습니다. 이를 위해 같은 문제를 여러 차례 반복 학습하고, 다른 접근 방식이나 변형된 문제에도 적용하며 학습하였습니다. 중간고사 및 기말고사 준비 과정에서는, 강의를 함께 듣는 다른 수강생들과 서로 모르는 부분에 대해 질문하고, 충분히 이해한 사람이 설명하는 반복 학습 구조를 적극 활용했습니다. 한 문제를 여러 사람의 시각으로 분석하고, 다양한 설명을 들으며 이해가 부족한 부분을 즉시 보완할 수 있었습니다. 또한 설명을 듣는 것에서 끝나지 않고, 다시 직접 문제를 풀어보며 완벽하게 이해할 수 있도록 하였습니다. 결과적으로, 단순하게 강의를 듣고 이해하는 것을 넘어, 스스로 사고하고 문제를 해결하며, 다른 사람에게 설명할 수 있는 수준까지 학습을 발전시킬 수 있었습니다. 이러한 반복적이고 능동적인 학습 과정을 통해 알고리즘의 개념을 정확하게 이해할 수 있었으며, 유사한 문제를 확인하였을 때, 적용할 수 있게 되었습니다.

강의 자료를 통해 알고리즘의 수학적 원리를 먼저 이해한 뒤에 R 코드로 구현해보고 알고리즘을 구현했습니다. 만들어낸 R 코드를 강의 자료의 문제에도 적용하고, AI에게 예시 문제를 만들어달라 한 후 여러 문제에 적용해보았습니다. 이해가 부족하거나 코드에 오류가 발생했을 때 AI에게 질문하고 피드백을 받으며 자기주도적으로 AI와 학습했습니다. 이러한 이론 학습과 실습, AI와 상호작용을 순환하는 과정을 통해서 복잡한 수식을 코드로 옮기고, 알고리즘의 작동 원리를 체득할 수 있었습니다.

통계적 알고리즘 수업에서 다양한 알고리즘들의 원리와 수학적인 배경을 학습하며, 이를 AI 튜터와 결합하여 학습의 효율을 증대시켰습니다. 또한, 학습한 알고리즘을 R 프로그램을 통해 직접 구현하는 과정을 통해 해당 알고리즘이 어떤 원리로 동작하는지 가시적으로 이해할 수 있었습니다. 알고리즘의 기본 바탕이 되는 수학적 배경의 학습이나, 알고리즘을 동작하기 위한 R 프로그램에서의 코드를 구현할 때 막히는 부분이나, 조금 더 정확하게 알고 싶은 내용들을 AI 튜터를 통해 질문하고 학습하며, 지금 이해한 내용이 맞는지 검증하는 과정을 통해 학습 효과를 증대시킬 수 있었습니다. 또한 매 주차 AI 튜터를 활용하여 도움이 된 부분을 프롬프트 질문과 AI 튜터의 답변을 기록하며, 학습한 개념을 돌아보고 이해가 되지 않는 부분들을 복습 과정에서 다시 한번 확인하며, AI 튜터를 통해 답을 구하며 알고리즘과 수학적인 배경, 코드들을 효율적으로 학습할 수 있었습니다. 하지만 AI 튜터의 답변이 구하고자 했던 답과 다른 경우엔 구글링 또는 유튜브 등과 같은 플랫폼의 강의 영상을 통해 부족한 개념을 이해할 수 있었습니다.

3. 본 강좌의 학습활동을 통하여 무엇을 배웠나?

수치해석과 선형대수 관련 다양한 개념과 알고리즘을 심도 있게 이해하고, 이를 실제 문제 해결에 적용하는 능력을 배웠습니다. 먼저 미분과 적분 관련 개념으로는 도함수, 고계 도함수, 수치 미분, 차분 근사, 정적분, 리만 합, 사다리꼴 공식, 심프슨 공식에 대하여 학습하며, 해석적으로 풀기 어려운 함수의 변화율이나 면적 계산을 수치적으로 하는 방법을 익혔습니다. 이를 통해 수학적 개념이 단순한 이론이 아니라 실제 계산 문제에 적용이 된다는 걸 직접 확인할 수 있었습니다. 또한 방정식과 연립방정식을 해결하는 방법에 대해서도 학습했습니다. 단일 방정식의 해를 구하기 위해 이분법, 뉴턴의 방법, 할선

법 등에 대해 학습하였고, 선형 연립방정식의 해를 구하기 위해 Jacobi 방법, Gauss-Seidel 방법, 켤레기울기법 등의 수치적 접근 방법을 이해하고 구현해 보았습니다. 이러한 과정에서 단순하게 공식만 달달 외우는 것이 아니라, 알고리즘의 원리와 반복 계산 과정을 직접 실행하며 결과를 이해하고, 다른 문제에도 직접 적용하는 경험을 할 수 있었습니다. 선형회귀와 최적화 관련 학습도 진행하였습니다. 정사영, 정규방정식, 경사하강법, 뉴턴의 방법, Quasi-Newton Method 등을 학습하며, 데이터와 모델을 수학적으로 연결하고 최적의 해를 찾는 과정에서 수치적 방법이 어떻게 사용되는지에 대해 학습할 수 있었습니다. 특히 경사하강법과 뉴턴 계열 방법을 비교하며 수렴 속도와 안정성의 차이를 직접 확인해 볼 수 있었습니다. 마지막으로 고윳값과 고유벡터 계산을 위한 Power Method와 Inverse Power Method를 학습하면서, 단순하게 행렬 계산을 넘어 데이터 분석과 차원 축소, 시스템 해석 등 다양한 실생활 문제에 적용 가능한 수치 해석적 접근이 어떻게 활용될 수 있는지를 배웠습니다. 이러한 여러 분야의 학습 활동을 통해, 수학적 개념과 알고리즘을 실제 문제에 적용하여 해결에 적용하는 능력, 복잡한 계산 과정을 단계별로 분석하고 이해하는 능력, 그리고 스스로 학습하며 문제를 해결하는 자기주도적 학습 능력까지 모두 키울 수 있었습니다.

새로운 문제가 주어졌을 때 왜 이 알고리즘을 적용해야 하는지, 그리고 R로 어떻게 구현해야 하는지를 알게 되었습니다. 복잡한 수학적 문제를 컴퓨터가 계산할 수 있는 알고리즘으로 변환하는 능력을 길렀습니다. 수치 해석으로는 사다리꼴, 심프슨 공식을 이용한 적분과 뉴턴의 방법을 이용한 방적식 풀이를 진행하였습니다. 효율성 측면으로는 역행렬을 직접 구하는 것보다 Gauss-Seidel이나 켤레기울기법 같은 반복법이 대규모 연산에서 효율적임을 배웠습니다. 최적화에서는 경사하강법과 뉴턴의 방법을 통해 손실 함수를 최소화하는 데이터 분석의 원리를 익혔습니다.

“통계적 알고리즘” 수업을 통해 기본적인 알고리즘에 대한 포괄적인 지식에 대해 학습할 수 있었습니다. 중간고사 이전 수업에서는 도함수, 고계 도함수, 수치 미분, 차분 근사, 정적분, 리만 합, 수치적분, 사다리꼴 공식, 심프슨 공식, 방정식, 방정식의 해, 이분법, 뉴턴의 방법, 할선법, 선형연립방정식, Jacobi 방법, Gauss-Seidel 방법 등과 같이 기본 알고리즘에 수학적 배경이 되는 기초 수학에 대해 학습하였고 이를 R을 통해 코드로 구현하며 가시적으로 해당 알고리즘이 어떻게 동작하는지에 대해 학습할 수 있었습니다. 이후, 정사영, 선형 회귀, 정규방정식, 경사하강법, 뉴턴의 방법, Quasi-Newton Method, 고윳값, 고유벡터, Power Method, Inverse Power Method 등과 같이 알고리즘을 조건과 상황에 따라 어떤 알고리즘을 선택해야 하며 알고리즘을 문제상황에 적용할 때 어떤 지점을 고려해야 하는지 배울 수 있었습니다. 또한, 수학적인 배경을 학습하거나 코드를 구현할 때 모르는 부분이 생겼을 때 AI 튜터와 같은 학습 보조 도구를 활용하며, 프롬프트를 작성해 보았고 답을 정확히 어떻게 찾아야 하는지, AI를 통해 구한 답이 맞는 것인지 검증해 보며 학습 보조 도구로써 AI를 활용하는 법을 익힐 수 있었습니다.

4. 다른 동료들로부터 무엇을 배웠는가?

다른 학생들이 질의응답 게시판에 작성한 질문과 AI 튜터의 답변을 살펴보면서, 같은 개념에 대해서도 학생마다 접근 방식과 질문의 관점이 다르다는 것을 확인할 수 있었습니다. 대부분의 학생들은 개념 위주로 설명해 달라는 형식의 질문을 작성했지만, 일부 학생들은 직접 내용을 요약하여 정리하고, 그 내용을 바탕으로 구체적인 질문을 덧붙이는 방식으로 학습을 진행하고 있었습니다. 이러한 요약 사례를 비교하면서, 제가 같은 개념을 정리할 때 놓쳤거나 다르게 접근한 부분을 파악할 수 있었고, 어떤 방식으로 개념을 요약하고 정리하는 것이 이해에 더 도움이 되는지를 구체적으로 확인할 수 있었습니다. 특히 기억에 남는 것은, 두 알고리즘의 장단점을 비교하거나 알고리즘을 적용할 때 주의해야 할 점을 포함하여 질문하고, 이를 요약해둔 사례였습니다. 처음에는 이러한 질문을 떠올리지 못했지만, 게시판에서 다른 학생들의 질문과 AI의 답변을 보면서 “단순히 개념을 이해하는 것에서 나아가, 알고리즘의 특

징과 활용상의 주의점을 고려한 질문을 작성할 수도 있구나"라는 깨달음을 얻을 수 있었습니다. 이를 통해, 학습 과정에서 단순히 강의를 듣고 개념을 외우는 것에 그치지 않고, 능동적으로 의문을 제기하고 자신의 이해를 점검하는 방법을 배울 수 있게 되었습니다. 또한 이러한 경험을 바탕으로, 이후 학습 활동에서는 저 역시 단순히 개념을 이해하는 수준을 넘어, 알고리즘의 성질과 장단점, 활용 상황 등을 고려하여 질문을 구성하고, 이를 기반으로 스스로 내용을 정리하고 요약하는 습관을 가지게 되었습니다. 이 과정을 반복하면서 단순 암기 중심의 학습에서 벗어나, 개념을 비교·분석하고 자신만의 이해 체계를 만들어가는 자기주도 학습 능력을 키울 수 있었습니다. 결과적으로, 다른 학생들의 질문과 답변을 관찰하고 비교한 경험은 학습 방법뿐만 아니라 사고의 깊이를 넓힐 수 있는 기회가 되었고, 알고리즘을 공부하거나 새로운 개념을 학습할 때 좀 더 전략적으로 접근할 수 있게 되었습니다.

동료들의 AI튜터 활용 방법을 공유하여 더 효과적으로 AI튜터에게 질문하거나 학습할 수 있었습니다. 같은 개념이라도 어떻게 프롬프트를 입력하느냐에 따라 답변의 깊이가 달라지는 것을 확인했고, 동료들의 질문과 방식을 참고하여 AI튜터를 주도적이고 효율적으로 활용할 수 있었습니다.
혼자 공부할 때 놓쳤던 개념들을 동료들의 질의응답을 통해서 생각해보지 못한 방향으로 개념을 다시 점검할 수 있었습니다.

다른 동료들의 복습 활동 작성 글을 통해 수업에서 배운 내용 중 어떤 부분이 미진한지 확인할 수 있었습니다. 다른 동료들의 복습 활동을 통해 수업 내외에서 알고리즘, 수학적 배경, 코드 등의 내용을 확인하며 미처 알고 있지 못한 부분을 학습할 수 있었습니다. 또한, 질문, 복습 내용, 추가적인 문제 풀이 등은 수업에서의 동기부여가 되었습니다. 다른 동료들의 학습활동을 통해 현재 학습 내용의 이해 수준을 진단하며 어떤 부분을 추가로 공부해야 하는지 갈피를 잡을 수 있었던 것 같습니다.

5. 새롭게 배운 내용을 실제 생활에 어떻게 적용할 것인가?

이번 "통계적 알고리즘" 수업을 통해 수학적, 통계적 정보들을 배경으로 한 알고리즘들을 단계적으로 이해할 수 있었습니다. 논문이나 레포트를 읽을 때 통계 기법을 사용하는 경우가 많지만, 해당 통계 기법, 알고리즘들이 어떤 의미를 함양하는지 이해하는 것이 어려움이 있었지만, 이번 "통계적 알고리즘" 수업을 통해 알고리즘에 대해 기본적인 정보를 이해할 수 있었습니다. 또한, 수학적인 문제들을 해결하는 과정에서 답을 구하기 어려운 조건, 상황일 때 가장 근사한 값을 구하기 위해 알고리즘들이 어떻게 동작하는지 실습 문제를 통해서 배울 수 있었고 이러한 경험을 통해 알고리즘을 사용하는 상황과 목적을 더 명확히 이해할 수 있게 되었습니다. R 프로그램을 통해 문제 상황을 직접 코드를 통해 해결하는 과정을 배우며, 단순히 코드를 실행하여 결괏값을 구하는 것을 넘어 해당 알고리즘의 수학적 배경이 어떻게 되는지 왜 해당 알고리즘을 사용해야 하는지 고려해야 함을 확인할 수 있었습니다. 또한, 이번 수업에서 사용하게 된 AI 튜터를 사용하며 효율을 극대화할 수 있을 것 같고 앞으로는 과제나 프로젝트에서 데이터 분석과 알고리즘을 사용할 때, 적절한 알고리즘을 선택하고 결과를 해석하는 과정에서 근거와 과정을 함께 제시하는 방식으로 실생활에 적용할 수 있을 것 같습니다.

수업에서 익힌 선형 회귀와 최소제곱법의 수학적 원리를 실제 데이터셋에 적용하여 변수 간의 상호작용을 규명하고 미래 추세를 예측하는 분석 프로젝트를 수행할 수 있습니다. 현상을 관찰하는 것을 넘어서 과거 데이터를 기반으로 정량적인 모델링을 수행함으로써 직감이 아닌 데이터에 근거한 객관적이고 논리적인 의사결정을 할 수 있을 것입니다. 더 나아가, 최적화 알고리즘에 대한 이해를 바탕으로 모델이 손실 함수를 최소화하며 최적의 가중치를 찾아가는 과정을 수학적으로 이해하고, 향후 프로젝트에서 모델의 성능을 개선하거나 문제를 해결하는 데에 주도적으로 응요할 수 있을 것입니다.

수치해석 및 선형대수 기반 알고리즘들은 나중에 전공인 데이터테크 및 임상의학통계를 공부할 때 실제 문제 해결 과정에서 폭넓게 적용할 수 있을 것이라 생각합니다. 특히나 임상 데이터는 연속적이면서도 이상치가 많고, 해석적으로 정확한 해를 구하기 어려운 경우가 많기 때문에, 본 강좌에서 배운 수치적 접근 방법들이 도움을 줄 수 있을 것으로 기대됩니다. 먼저, 도함수, 수치 미분, 차분 근사, 수치 적분 등은 임상 실험 데이터나 생체 신호 데이터 분석에 적용할 수 있습니다. 예를 들어 환자의 시간에 따른 생체 지표 변화율에 대해 분석하거나, 특정 기간동안의 누적 효과를 계산할 때 해석적 계산이 어려운 경우 수치적 방법을 활용해 근사값을 구할 수 있습니다. 이는 약물 반응 분석이나 질병 진행 속도 평가 등 실제 임상 연구 상황과도 밀접하게 연결됩니다. 또한 방정식의 해를 구하는 방법은 통계 모델의 파라미터 추정 과정에서 활용할 수 있습니다. 임상의학통계 분야에서는 손실 함수를 최소화하는 과정이 자주 등장하는데, 이때 닫힌 형태의 해가 존재하지 않는 경우 수치적 반복 알고리즘을 통해 근사해를 구해야 합니다. 이때, [통계적 알고리즘] 강의에서 배운 방법들이 이러한 최적화 문제를 이해하고 구현하는 데 중요한 기초가 될 것 같습니다. 선형연립방정식, Jacobi 방법, Gauss-Seidel 방법, 켤레기울기법은 대규모 임상 데이터 분석이나 고차원 통계 모델을 다룰 때 특히 유용합니다. 데이터테크 전공에서 다루는 빅데이터 환경은 계산량이 매우 크기 때문에, 효율적인 반복 해법을 선택하는 것이 중요합니다. 이러한 알고리즘을 통해 계산 효율성과 수렴 특성을 고려한 분석이 가능해질 것이라고 생각합니다. 나아가 정사영, 선형 회귀, 정규방정식, 경사하강법, 뉴턴 및 Quasi-Newton Method는 임상의학통계의 핵심 분석 기법과 직접적으로 연결됩니다. 환자 특성과 질병 결과 간의 관계를 모델링하거나, 예측 모델을 구축하는 과정에서 이러한 알고리즘들은 기본적인 이론적 토대가 됩니다. 특히 경사하강법과 뉴턴 계열 방법을 이해함으로써, 통계 패키지나 머신러닝 모델이 내부적으로 어떻게 학습되는지에 대해 깊이 있게 이해할 수 있을 것이라 생각했습니다. 마지막으로, 고윳값, 고유벡터, Power Method, Inverse Power Method는 차원 축소나 데이터 구조 분석에 활용할 수 있습니다. 임상 데이터는 변수의 수가 많고 상관관계가 복잡하다고 알고 있는데, 주요 정보를 요약하고 해석하는 과정이 중요하다고 들었습니다. 이러한 기법들은 주성분 분석과 같은 방법의 기반이 되며, 고차원 데이터를 효율적으로 해석하는 데 실질적인 도움이 될 것이라 생각됩니다. 종합적으로, 이번 수업에서 배웠던 내용들은 단순한 이론 학습에 그치지 않고, 제가 선택한 전공인 데이터테크 및 임상의학통계에서 다루게 될 실제 데이터 분석과 연구 문제를 이해하고 해결하는 데 중요하게 적용될 것이라 생각됩니다. 만약, 추후에 전공 등에서 수치적 계산과 알고리즘 선택이 필요한 상황이 발생했을 때, 이러한 내용들을 최대한 적극적으로 적용해 나가고자 합니다.

2장. 참여 부분

1. 다음 질문에 대하여 간단히 서술하세요!

(1) QnA 게시판 참여 횟수 <QnA에서 직접 확인하세요>: 12월 12일까지의 참여 기록

총 __**__회 (글작성: __**__회, 댓글: __**__회)

(2) 다음 밑줄 친 곳에 들어갈 내용을 고르시오.

나는 _____의 개념을 이해하고 (간단한 것은 손으로, 복잡한 것은 R을 이용하여) 계산할 수 있다.

도함수, 고계 도함수, 수치 미분, 차분 근사, 정적분, 리만 합, 수치 적분, 사다리꼴 공식, 심프슨 공식, 방정식, 방정식의 해, 이분법, 뉴턴의 방법, 할선법, 선형연립방정식, Jacobi 방법, Gauss-Seidel 방법, 켤레기울기법(Conjugate Gradient Method), 정사영, 선형 회귀, 정규방정식, 경사하강법, 뉴턴의 방법, Quasi-Newton Method, 고윳값, 고유벡터, Power Method, Inverse Power Method

(위 Box 안의 키워드에서 자신이 충분히 이해하고 있는 것을 표시하면 됩니다.)

2. AI 튜터(https://tutor.k-university.ai/)를 사용한 소감을 간단히 적으면 됩니다. (본인의 학습에 도움이 된 부분, 개선이 필요한 부분 등)

이번 “통계적 알고리즘” 수업을 통해 AI 튜터를 사용하며, 공부하며 막히던 부분에 대해 즉각적으로 질문에 대한 답변을 받을 수 있다는 점이 굉장히 도움이 되었습니다. 수업 특성상 막히는 부분을 이해하지 못하고 넘어가게 될 때 눈덩이처럼 이해하지 못한 내용들이 연결되어 학습을 부진하게 만들곤 했었는데 이번 AI 튜터를 사용하며, 이해하지 못한 개념이나 정보를 즉시 확인할 수 있어 진도를 따라가는 데 큰 도움이 되었습니다. 다만, AI가 사용자의 의도를 벗어난 답변을 하거나, 부정확한 답변을 하는 경우 또한 존재하였습니다. 이런 경우 AI를 사용하기 위한 적절한 프롬프트 예시 문항 제공하는 등의 도움이 있다면 학습 상황에서 사용하기 더 좋을 것 같다고 생각하였습니다.

AI튜터를 이용한 학습은 단순 검색이 아닌 같이 학습할 수 있는 도구였습니다. 강의 내용 복습 중 헷갈렸던 ‘선형 회귀의 기준’이나 ‘알고리즘의 수렴 조건’ 등에 대해 꼬리에 꼬리를 물며 깊게 파고드는 질문을 던지면서, 동료들에게도 물어보기 꺼렸든 질문들을 개인적으로 편하게 물어보고 이론을 학습할 수 있었습니다. 다만, 가끔 질문의 의도와 다르게 너무 교과서적인 답변만 반복하는 경우가 있어, 문맥을 더 잘 파악하여 맞춤형 예시를 들어주는 기능이 강화되면 좋겠습니다."

강의를 듣다가 이해가 가지 않는 코드나 강의안 내 이해가 어려운 부분이 있을 경우, AI 튜터를 활용하는 방식을 가장 많이 사용했습니다. 수업 시간에 배운 여러 알고리즘 중에서 이해가 가지 않는 부분을 구체적으로 질문함으로써, 해당 부분의 궁금증을 해결할 수 있었고, 이를 통해 많은 문제를 AI 튜터를 통해 해결할 수 있었습니다. 또한, 수업에서 다룬 개념 외에 추가로 더 궁금한 사항, 예를 들어 어떤 알고리즘을 특정 상황에서 더 적합하게 사용할 수 있는지, 알고리즘 간의 차이점은 무엇인지, 알고리즘을 사용할 때 주의해야 하는 점은 무엇인지 등에 대해 질문함으로써, 얄팍했던 알고리즘 개념을 깊이 있게 쌓을 수 있었습니다. 특히, 이해가 어려운 알고리즘을 예시를 들어 단계별로 설명해 달라고 프롬프트를 작성하면, AI 튜터가 차근차근 단계별 응답을 제공해 주었으며, 이를 통해 알고리즘을 점차 이해할 수 있었습니다. 그러나 AI 튜터에도 몇 가지 한계점이 존재했습니다. 예를 들어, “켤레 기울기법을 R에서 쓸 수 있는 코드로 알려줘”라고 질문했을 때, 질문 의도와 다르게 관련 없는 코드나 답변이 제공되는 경우가 있었습니다. 질문의 핵심과 관련 없는 단어에 초점을 맞춰 답변이 나오거나, 질문의 의미와 반대되는 답변이 나오기도 하여, 원하는 답변을 얻기 위해 여러 차례 질문을 수정해야 하는 번거로움이 있었습니다. 또한, [통계적 알고리즘] 과목의 과제 중 AI 튜터의 답변을 정리해 제출하는 과제가 있었는데, 개념을 묻거나 해당 개념의 예시를 요청한 경우, AI 튜터는 종종 수식 형태로 답변을 제공했습니다. 그러나 해당 답변을 복사하여 스마트리드에 붙여넣으면 수식이 제대로 표시되지 않거나, R에서 바로 사용할 수 있는 함수 형태로 변환되지 않아 불편함이 있었습니다. 따라서 AI 튜터가 수식을 제공할 때, 복사 후에도 R 등에서 바로 활용할 수 있는 형태로 제공될 수 있다면 더욱 효율적으로 과제를 수행할 수 있을 것이라고 느껴 이 부분이 개선되었으면 좋겠다고 생각했습니다.

AI 튜터는 복잡한 수식의 유도 과정(예: 정규방정식의 도출이나 테일러 급수를 통한 오차 증명)을 단계별로 풀어 설명해 주어, 전공 서적만으로는 이해하기 힘든 부분을 빠르게 보완할 수 있었다. 또한, R 코드를 작성할 때 단순히 정답 코드만 던져주는 것이 아니라, '왜' 그 방식이 더 효율적인지(예: solve(A)보다 solve(A, b)가 수치적으로 안정적인 이유)를 설명해 주어 코딩 습관을 교정하고 알고리즘의 본질을 이해하는 데 큰 도움이 되었다. 또한 매주 복습을 통해 주차별로 학습을 상기시켜 이해를 도왔다.

AI 튜터를 활용하면서 수업 중 바로 이해되지 않았던 개념을 추가로 질문하고 정리하는 데 많은 도움이 되었다. 특히 예시나 단계별 설명을 통해 알고리즘의 흐름을 이해하는 데 유용했다. 다만 답변의 난이도가 다소 높게 느껴질 때도 있어, 기본 개념부터 차근차근 설명해 주는 기능이 더 보완되면 좋겠다.

수업 시간에 이해하지 못하거나 놓쳤던 개념들을 AI를 통해 다시 확인하며 학습할 수 있었습니다. 특히 알고리즘의 계산 과정과 흐름을 차분히 정리해 볼 수 있어 수업 내용을 보완하는 데 도움이 되었습니다. 이를 통해 수업에서 배운 내용을 보다 안정적으로 이해할 수 있었다고 느꼈습니다.

도움이 된 부분은 AI 튜터로 통계적 알고리즘수업에서 배운 내용을 복습하고 이해하는 데 큰 도움이 되었습니다. 특히, R 프로그래밍을 활용한 알고리즘 구현 과정에서 발생하는 문제들을 즉각적으로 해결할 수 있었습니다. 개선이 필요한 부분으론 AI 튜터가 제공하는 정보가 너무 일반적이어서 깊이 있는 설명이 부족할 때가 있었습니다. 구체적인 예제나 코드가 포함된다면 더욱 유용할 것 같습니다.

ai튜터를 활용 강의 내용을 이해하는 데 큰 도움이 되었다. PDF 자료만으로는 이해하기 어려웠 부분을 질문을 통해 이해할 수 있었다. 특히 반복 알고리즘이나 행렬 관련 개념들은 한 번에 이해하기 힘들었었는데 ai튜터를 통해 이해하는데 많은 도움이 되었고 ai튜터가 설명을 해줄 때 단순히 답을 알려주는 것이 아니라 개념의 흐름과 의미를 함께 설명해줘 이해하고 학습하는데 효과적이였던 것 같다. 이를 바탕으로 다음 내용을 배우는데 좀 더 수월하게 배울 수 있었다.

■ “통계적알고리즘” PBL 보고서를 마치고 (최종 소감)

[통계적 알고리즘] 강의를 처음 수강했을 때는 전까지 많이 접하지 못했던 수치해석과 알고리즘 중심의 내용이 다소 어렵게 느껴졌고, 단순한 개념 이해만으로는 수업 내용을 완전히 따라가기 힘들다고 느꼈습니다. 특히 알고리즘의 흐름과 수식, 그리고 이를 코드로 구현하는 과정이 익숙하지 않아 학습 초반에는 많은 어려움과 시행착오를 겪었습니다. 그러나 강의안을 반복해서 정리하고, 이해되지 않는 부분은 AI 튜터와 질의응답 게시판을 적극적으로 활용하며 학습을 이어간 결과, 점차 알고리즘의 원리와 목적을 이해할 수 있게 되었습니다. 단순히 문제를 푸는 데서 그치지 않고, 알고리즘의 장단점과 적절한 적용 상황을 고민하고 학습하면서 개념에 대한 이해가 한층 깊어졌다고 느꼈습니다. 또한 다른 학생들의 질문과 요약 내용을 참고하며, 스스로 학습 내용을 구조화하고 정리하는 방법을 배울 수 있는 계기가 되었습니다. 현재는 강의를 배우기 전과, 강의를 처음 배웠을 때에 비해서 훨씬 안정적으로 강의 내용을 이해할 수 있게 되었으며, 배운 알고리즘들이 단순한 이론이 아니라 데이터 분석과 임상의학통계와 같은 전공 분야에서 실제로 활용될 수 있는 중요한 도구라는 점을 인식하게 되었습니다. 특히 수치적 최적화, 선형 회귀, 고윳값 계산과 같은 내용들은 향후 전공 수업이나 연구 과정에서도 핵심적인 역할을 할 것이라고 느끼게 되었습니다. 이번 강좌를 통해 저는 단순히 새로운 지식을 습득한 것을 넘어, 어려운 개념을 스스로 해결하고 이해해 나가는 학습 태도와 문제 접근 방식을 기르는데 많은 도움을 얻었습니다. 앞으로도 [통계적 알고리즘]에서 배운 내용과 학습 방법을 바탕으로, 전공 학습과 실제 데이터 분석 상황에서 더욱 능동적으로 문제를 해결해 나가고 싶습니다.

통계적 알고리즘 보고서를 마무리하면서 지난 학기 동안의 학습 내용과 과정을 차분히 돌아보니, 이론만이 아니라 실제 문제를 해결하는 능력도 크게 늘었다는 점을 느꼈다. 이 수업은 단순히 공식을 외우는 데 그치지 않고, 같은 문제를 다양한 방식과 여러 R 코드로 접근할 수 있다는 것을 제대로 알게 해주었다. 가장 기억에 남는 내용은 선형연립방정식의 반복해법에 대해 배웠던 시간이다. 겉보기엔 비슷해 보여도 이분법, 뉴턴의 방법, 할선법 각각이 수렴 속도나 미분 필요 등에서 뚜렷한 차이와 장단점을 지니고 있음을 깨달았다. 문제 상황에 따라 이들 중 어떤 방법을 선택할지 고민하는 과정에서, 수치 계산법은 단순히 계산을 대신해주는 데 그치지 않고, 효율성과 안정성까지 함께 고려해야 한다는 걸 알게 됐다. AI 튜터를 적극적으로 활용한 경험도 인상 깊었다. 평소 이해가 잘 가지 않았던 Jacobi 방법이나 켤레기울기법 같은 개념과 차이점에 대해 바로 질문할 수 있었고, 즉각적인 답변 덕분에 헷갈렸던 부분을 바로잡거나, R 코드로 직접 구현해보며 여러 문제를 해결해볼 수 있었다. 복습할 때도 튜터와 함께 하면서 그날 배운 개념들을 한 번 더 정리할 수 있었고, 질문 내용을 고민하고 답을 찾는 과정에서 스스로도 많이 성장할 수 있었다. 매주 질의응답 시간을 통해 학습한 내용을 다시 생각해보고 서로의 의견을 공유하면서, 동료들이 문제를 바라보는 다양한 시각도 접할 수 있었다. 내가 정리한 내용이 다른 친구들에게도 도움이 됐다는 이야기를 들을 때마다, 뿌듯한 마음으로 한 주를 마무리할 수 있었다. 무엇보다 이 수업은 익숙했던 내용을 더 깊이 배우고, 새로운 주제도 폭넓게 익힐 수 있는 소중한 기회였다. 중간에 어렵다고 포기하고 싶은 순간도 있었지만, 수업 외에도 스스로 문제를 풀고 다시 정리해보면서 끝까지 해낼 수 있었다. 이제는 배운 것들을 바탕으로 앞으로 더 복잡한 시스템이나 문제를 만나도, 가장 효율적인 방법을 찾아 스스로 해결해 볼 수도 있겠다는 자신감도 갖게 됐다.

이번 수업을 들으면서 여러가지 작동하는 코드들을 직접 만들면서 다양하게 학습하였습니다 처음에 코드 만드는 것이 정말 어렵고 사실 주차를 거쳐가면서도 어려웠던 것 같습니다. 하지만 그런 코드들을 보면서 알고리즘이 어떤 식으로 작동하는지 어떤 때에 쓰는지 알 수 있는 그런 수업이였던 것 같습니다. 또한 ai튜터를 쓰면서 혼자 공부를 하는 것이 아니고 막히는 부분에서 물어볼 것이 있다는 게 도움이 많이 되었습니다. 다음에 또 좋은 과목이 있으면 수강하면 좋을 것 같습니다.

R을 그저 코딩이라고 생각했는데 여러 수학적 요인과 방법들도 배우면서 수학을 코딩으로 옮겨서 할 수

있다는 점이 흥미롭고 수학과 코딩을 동시에 배우며 해결할 수 있다는 점이 유익했습니다. 지난 과제들을 보면서 한 학기동안 많은 것들을 배웠구나하는 생각도 들고 개념만 배운다는 느낌보다는 이를 활용하고 비교하면서 알맞게 필요한 상황에 어떤 방법을 써야하는지도 배우게 되는 수업이었습니다.

통계적 알고리즘을 수업하며 학기말이 되고 수업이 끝날때 쯤에는 수업에 대한 내용을 전반적으로 이해할 수 있게 되었으면 좋겠다라는 생각을 가지고 수업에 임하였습니다. 처음에 알고리즘을 기반으로 수학적 공식을 학습할 때에는 어려움이 있어 주변 친구들과 AI튜터에 도움을 받아 공부하였지만, 한 번 이해하고 난 뒤로부터 학습하는 과정을 이해하고 방법을 찾아갈 수 있었고 문제 해결 방법도 점차 스스로 찾아갈 수 있게 되었습니다. 또한 어려웠던 공식을 그냥 넘어가지 않고 처음부터 되돌아보는 습관을 기를 수 있게 되었습니다. 이를 통해 R 코드에 오류가 났을때 두려워한다기보다 어떤 부분에서 오류가 났는지 살펴보고 해결하는 과정에 익숙해질 수 있었고 처음부터 완벽히 이해해야 한다는 부담감을 줄일 수 있었습니다. PBL 보고서를 쓰며 한 학기동안 한 학습활동을 되돌아 보았을때 한 학기동안 배운것도 많고 얻어간것도 많았다고 생각합니다. 동료들에게 도움을 받을 때 나도 개념을 완벽히 숙지하지 못하더라도 아는 개념은 잘 설명할 수 있을때까지 학습을 해야한다고 생각하여 보고서를 쓰며 과제를 했던 것에 대한 보충설명을 적을 때 나만의 언어로 적으려고 많이 노력했습니다. 그 결과 시험공부를 하며 아는 개념을 충분히 설명할 수 있는 정도가 되어 스스로 뿌듯했던 경험이 있습니다. 하지만 코드 학습 부분에서 스스로 부족한 부분이 있었는데 그 부분에서는 완벽하게 학습이 덜 된 것 같아 아쉬움이 남습니다. 질의응답 게시판에 올라온 다른 수강생분들의 과제를 보며 나와는 다른 부분이 중요하다고 생각하는 부분에서 내가 미처 알지 못하였던 부분과 짚고 넘어가지 못했던 부분을 다시 한 번 더 되짚어 볼 수 있는 기회가 되었고, 시험공부를 할 때 더 폭 넓게 공부를 할 수 있었습니다. 보고서를 쓰면서 그동안 내가 학습한 부분을 돌아보는 것이 얼마나 중요한지 알게되었습니다. 그 전에는 복습을 할 때 아는 것은 그냥 넘어가는 경우가 대다수였어서 막상 시험장에 들어가면 다 안다고 생각했던 것들이 생각이 나지 않는 경우가 많았지만 복습을 하고 정리를 하며 학습하게 되니까 쉬운 개념은 물론이고 어려워서 헷갈려했던 개념을 완벽하게 알고 갈 수 있게 되어 시험장에서 문제를 푸는 속도가 전보다 더 빨라진 것을 느낄 수 있었습니다. 이러한 기회로 원래라면 오류가 나는 R코드를 따로 정리한다기 보다는 개념을 이해하는 것으로 끝났을텐데 R코드에서 학습이 덜 되었던 부분을 따로 시간을 내어 학습하는 습관을 기를 수 있게되었습니다. 알고리즘을 R로 작성하는 부분에서 수학 공식을 이해하지만 R로 작성할 수 없었다는 것에 처음에는 겁을 먹고 학습을 제대로 하지 못하였습니다. 그럴때면 동료들에게 도움을 받아 이해를 하는 것을 목적으로 두고 학습하였는데 그렇게 하다보니 나만의 것으로 만드는 것에는 시간이 많이 걸린다는 것을 알게 되어 스스로 학습하는 습관을 길렀습니다. 특히 부족하다고 생각했던 알고리즘을 R로 작성하는 부분에서는 교수님께서 알려주신 코드를 바탕으로 학습을 하고 모르는 부분은 AI튜터의 도움을 받아 작성하였습니다. 튜터의 도움을 받았지만 이해가 되지 않는 부분은 따로 정리하고 수학 공식을 다시 한 번 살펴보는 식으로 공부를 하였습니다. 저는 처음에 다짐한대로 수업이 끝날 때 쯤 수업 내용을 전반적으로 이해하는 것을 목표로 둔 것을 보고서를 쓰며 그 목표가 이루어지고 있다는 것을 깨닫게 되었습니다. 처음에는 수업에 대한 막연한 두려움이 있었다면 교수님과 동료들 그리고 AI 튜터를 통해 공부하다보니 못할것도 할 수 있게 된다는 것을 알게되어 공부를 할 때 스스로 자신감을 많이 가질 수 있게 되었습니다.

통계적 알고리즘 강의를 듣기 전 지난 학기에 교수님의 '데이터 분석의 수리적 기초'와 '행렬과 데이터 분석' 강의를 수강하며 데이터 과학에 필요한 수학적 지식들을 단단히 다질 수 있었습니다. 팀원들과 이론을 이용하여 프로젝트도 진행해 보고, SageMath를 이용하여 직접 계산해보기도 하였습니다. 하지만 수학적 이론들이 실제 컴퓨터 환경에서 어떻게 변형하고 사용되는지 의문이 있었습니다. 이번 '통계적 알고리즘' 강의에서 저에게 친숙한 R 프로그래밍을 통해 직접 구현해 봄으로써, 데이터 분석 역량을 완성하고자 수강하게 되었습니다. 강의 초반, 미분과 적분을 컴퓨터가 계산할 수 있는 '차분 근사'나 '사다리꼴 공식'으로 직접 구현해 보면서, 수학 이론이 어떻게 컴퓨터의 언어로 번역되어 계산 알고리즘이 되는지 경험해보았습니다. 평소 데이터를 시각화할 때에만 사용하던 R을 수학적인 알고리즘을 만들어 적용시키는 활용 방

법은 저에게 낯선 경험이었습니다. 단순히 공식으로 배웠던 선형 회귀가 실제로 데이터와 모델 사이의 오차를 최소화하는 '최소제곱문제'이고, 이를 해결하기 위해 행렬을 이용한 정규 방정식을 풀거나 경사하강법을 사용하여 최적의 파라미터를 찾아간다는 과정을 직접 코드로 구현하고 성공적으로 작동하였을 때에 성취감을 느꼈습니다. 또한 새로운 모델을 배우기 전 도함수, 고계도함수, 수치 미분/적분, 리만 합, 차분근사, 방정식, 선형 연립방정식, 정사영, 경사하강법, 양의 정부호, 고윳값, 고유벡터 등 필요한 수학적 도구들을 먼저 익혔습니다. 처음에는 연관성이 없는 개별적인 수학 이론들과 지식들이 실제 모델의 작동 원리를 학습하는 과정이나 알고리즘을 구현하는 과정에서 퍼즐 조각들이 깔끔하게 맞아 떨어지듯이 개념들이 쓰이는 방식에 즐거움을 느꼈습니다. 이를 통해 기초 이론이 얼마나 중요한지 깨닫게 되는 시간이기도 했습니다. AI 튜터를 통한 학습은 새로운 학습 방법과 태도를 변화시켰습니다. 기존에는 AI에게 단순히 정답만을 물어봤었다면, 현재는 "왜 이 알고리즘이 수렴하지 않는지.", "선형 회귀와 비선형 회귀를 가르는 기준이 무엇인지.", "수학적 이론이 어떻게 도출되는지." 등 깊이 있는 질문과 피드백을 받는 과정을 반복하여 강의에서 학습한 내용들 뿐만 아니라 능동적인 탐구를 할 수 있는 학습 태도를 갖게 되었습니다. 지난 학기에 배운 수리적 기초와 행렬 이론과 이번 학기 배운 통계적 알고리즘 구현 능력에 대한 학습은 저에게 데이터를 바라보는 관점과 분석 능력을 증진시킬 수 있는 시간이었습니다. 이 강의들 이후로 새로운 데이터나 낯선 문제에도 수학적 원리를 파악하고 알고리즘적으로 접근하여 해결해 볼 수 있는 능력이 생겼습니다. 강의 외적으로도 문제를 바라보는 관점과 그 문제를 해결하기 위해 정답을 구하는 것이 아닌 원리를 파악하고 탐구하여 해결해 나갈 수 있는 학습 능력을 높일 수 있었습니다. 한 해 동안 배운 원리와 활용 방법들이 앞으로 마주할 전공 수업들을 이해하는 데에 큰 도움이 될 것 같습니다. 감사합니다.

보고서를 쓰면서 생각한 것보다 학습한 내용이 많은 것 같습니다. 이것들을 어떻게 이해를 했을까 라는 생각이 많이 들었습니다. 강의를 들으면서 복습을 진행하고 이해한 내용을 정리하면서 질의 응답에 올리며 복습하고 다른 학우들이 올린 것들을 보면서 다시 복습하는 과정이 많이 도움이 된거 같습니다. 확실히 이해가 안된다고 포기하는 것이 아니라 반복 학습을 통해서 이해할때까지 노력을 해야한다는 것을 배웠습니다. 강의를 통해서 통계를 할때 어떠한 알고리즘으로 진행이 되는지도 많이 배웠습니다. 이후에 함수에 대해 이건 어떻게 진행이 되는 것이냐 라고 누군가가 물어본다면 잘 설명해줄 수 있을 거 같습니다.

이번 학기 통계적 알고리즘 수업은 단순히 주어진 공식을 외워서 문제를 푸는 것을 넘어, 데이터를 다루는 도구(알고리즘)이 실제로 어떻게 작동하는가?에 대한 근본적인 물음에 답을 찾는 과정이었다. 학기 초반에는 R의 문법과 기초 미적분학 개념이 다소 낯설게 느껴졌지만, AI 튜터와의 끊임없는 문답을 통해 수학적 추상 개념을 구체적인 코드로 구현해 보며 점차 자신감을 얻었다. 특히 최소제곱문제를 선형대수학의 정사영개념으로 해석하고, 최적화 문제를 해결하기 위해 경사하강법과 뉴턴법을 직접 바닥부터 구현해 보면서, 통계학의 이론들이 실제 컴퓨터 안에서 어떻게 계산되고 최적화되는지 투명하게 들여다볼 수 있었다. 또한, 이론적으로 완벽해 보이는 알고리즘도 컴퓨터의 연산 오차나 초기값 설정, 계산 비용의 문제로 인해 실패할 수 있음을 배우며, 데이터 분석가로서 상황에 맞는 최적의 알고리즘을 선택하고 튜닝하는 안목을 기를 수 있었다. 앞으로 마주할 복잡한 데이터 분석 문제에서도, 이번 수업에서 배운 수치적 사고방식(오차, 수렴, 효율성)을 바탕으로 문제의 본질을 꿰뚫어 보고 최적의 해결책을 찾아내는 학생이될 것이다. 한 학기 동안 열정적으로 지도해 주신 교수님께 감사드립니다!

통계적 알고리즘 PBL 보고서를 마치면서 한 학기 동안 배운 내용을 종합적으로 돌아볼 수 있었습니다. 수치미분과 수치적분부터 시작해 선형연립방정식, 반복해법, 최적화 알고리즘, 고윳값 계산까지 이어지는 과정이 처음에는 매우 어렵게 느껴졌지만, 점차 서로 연결된 개념이라는 점을 이해하게 되었고 복습을 몇번 혼자 더 해보니 이해하는 과정이 크게 힘들지 않았던 것 같습니다. 특히 AI 튜터를 활용해 질문하고 연습 문제를 만들어 풀어본 경험이 학습 내용을 정리하는 데 큰 도움이 되었습니다. AI를 활용하는 것 보다는 혼자 머리를 싸매는 과정이 길었었는데, AI를 적절히 활용하는 것이 오히려 학습에 도

움이 되었습니다. PBL 활동을 통해 단순히 이론을 배우는 데 그치지 않고, 알고리즘의 원리와 활용 목적을 함께 생각해 볼 수 있었습니다. 이번 과정을 통해 스스로 학습하는 방법에 대해서도 많이 고민해 볼 수 있었고, 앞으로 다른 전공 과목이나 실습에서도 이러한 학습 방식을 계속 활용할 것 같습니다.

본 과목을 수강하며 다양한 통계적 알고리즘과 수치적 방법을 학습하는 과정에서, 단순히 알고리즘의 계산 절차를 익히는 것을 넘어 “이 알고리즘을 언제, 어떤 문제에서, 어떤 기준으로 사용해야 하는가”에 대한 질문이 점차 중요해졌다고 느꼈다. 이는 여러 주차에 걸쳐 반복적으로 등장한 알고리즘들이 서로 다른 문제를 푸는 것처럼 보이면서도, 근본적으로는 유사한 문제 해결 구조를 공유하고 있었기 때문이다. 강의 전반에서는 수치 미분, 반복 해법, 최적화, 고윳값 문제, 선형 연립 방정식 등 다양한 알고리즘이 제시되었으며, 각 알고리즘에 대해 명확한 수학적 정의와 반복식, 수렴 조건, 허용 오차, 초기값 설정 등이 함께 다루어졌다. 이를 통해 알고리즘이 성립하기 위한 전제 조건과 계산 과정에서 발생할 수 있는 한계, 예를 들어 학습률 선택에 따른 발산 가능성, 초기값에 대한 민감성, 계산 비용 증가 등의 문제를 구체적으로 이해할 수 있었다. 이러한 내용은 알고리즘을 무조건적으로 적용하는 것이 아니라, 문제의 성격과 조건을 고려해야 함을 지속적으로 환기시켜 주었다. 특히 중반부 이후의 강의에서는 방정식의 해를 구하는 문제가 최적화 문제로 재해석되고, 경사하강법, 뉴턴 방법, 준뉴턴 방법, Conjugate Gradient 방법 등이 하나의 공통된 틀 안에서 설명되었다. 이를 통해 서로 다른 알고리즘들이 독립적인 기법이 아니라, 목적 함수 최소화라는 동일한 문제를 서로 다른 정보 수준과 계산 전략으로 해결하고 있음을 이해하게 되었다. 또한 고윳값과 고유벡터, PCA, Power Method 등의 내용에서는 모든 값을 정확히 계산하기보다, 문제 해결에 필요한 정보에만 수렴하는 반복 해법의 중요성이 강조되었다. 다만 강의자료는 이러한 알고리즘들이 실제 데이터 분석이나 모델링 과정에서 “어떤 상황에서 선택되어야 하는지”를 명시적으로 비교·정리하기보다는, 수식과 조건, 실습 문제를 통해 학습자가 스스로 판단하도록 구성되어 있었다. 즉, 알고리즘의 사용 맥락이 직접적으로 설명되기보다는, 수렴 조건과 계산상의 제약, 반복 구조를 통해 간접적으로 제시되었다고 볼 수 있다. 이로 인해 알고리즘의 이론적 구조는 명확히 이해할 수 있었으나, 실제 문제 해결 흐름에서의 위치와 선택 기준에 대해서는 추가적인 사고와 정리가 필요하다고 느꼈다. 이러한 경험을 통해 본인은 알고리즘 자체보다 중요한 것은, 주어진 문제를 어떻게 해석하고 어떤 수치적 방법이 그 해석에 적합한지를 판단하는 사고 과정이라는 점을 학습을 통해 확인할 수 있었다. 본 과목은 이러한 판단 능력을 수식과 반복 구조를 통해 훈련하도록 설계된 강의였으며, 이를 바탕으로 이후 실제 데이터 분석이나 모델링 문제에서도 알고리즘의 목적과 한계를 고려하며 적용할 수 있는 기초를 마련했다고 생각한다.

PBL 보고서를 쓰며 다른 강의도 이렇게 기록으로 남겨야겠다는 생각을 했다. 그리고 지금까지의 기록을 남기지 못한 것이 후회스러웠다. 4학기를 끝내면서 지금까지 내가 어떤 것을 배웠는지 정확히 알지 못해 답답함을 느끼는 경우가 많았기 때문이다. 내가 뭘 알고 모르는지 명확하지 않으니, 앞으로의 방향을 잡기가 어려웠던 것이다. 한학기 동안 배운 것을 정리하며 내용을 다시 들여다보게 되어 저절로 복습도 되었다. 방학 때 어디를 더 보충해야 할지 보이기도 한다. 1장의 질문에서 나 자신에게 학습 이해와 학습 방법, 동료들에게서 얻어간 것들과 배운 내용을 어떻게 실제 생활에서 써먹을 건지에 대해 물어보게 되어 자기 반성이 되었다. 보고서를 쓰며 생각보다 나는 학습 방법과 동료들과의 교류, 앞으로 이 내용을 어떻게 활용할 것인지에 대한 성찰이 부족하다고 느꼈기 때문이다. 다음 학기와 이번 방학때는 더 깊게 생각하며 공부에 열심히 임해야 할 것이다. 종합적으로 PBL보고서를 쓰며, 교수님이 말씀하신대로 나중에 면접을 보게 되든 공부를 깊게 해야하든 기록으로 남기는 것은 큰 도움이 된다는 것을 깨달았다.

이번 통계적 알고리즘 수업을 통해 수학적 개념과 알고리즘을 단순히 공식으로 익히는 것이 아니라, 실제로 어떻게 작동하고 어떤 상황에서 활용되는지를 중심으로 학습할 수 있었다고 생각합니다. 수업 초반에는 도함수, 수치 미분, 적분과 같은 기본적인 개념부터 시작하여, 점차 방정식의 해를 구하는 방법과 최적화 알고리즘, 선형대수 기반의 반복 알고리즘까지 다양한 내용을 다루면서 학습의 깊이가 점점 확장되는 것을 느낄 수 있었습니다. 특히 뉴턴의 방법, 할선법, 경사하강법, Quasi-Newton Method 등과 같은 알고리즘을 배우면서 각 방법이 어떤 이론적 배경을 가지고 있으며, 계산 과정에서 어떤 정보

를 활용하는지에 대해 고민해 볼 수 있었습니다. 단순히 하나의 방법을 익히는 데 그치지 않고, 문제의 특성과 계산 환경에 따라 어떤 알고리즘이 더 적합한지 비교하고 판단하는 과정이 인상 깊었습니다. 처음에는 수식과 행렬 계산이 부담스럽게 느껴졌지만, 반복적으로 문제를 접하고 예제를 따라가면서 알고리즘의 흐름을 중심으로 이해하려고 노력하게 되었습니다. 이 과정에서 수업 시간에 놓친 개념이나 이해가 부족한 부분은 AI를 활용해 다시 정리하며 학습을 보완할 수 있었습니다. 단순히 강의를 듣는 것보다 직접 내용을 정리하고 설명하는 과정에서 이해가 훨씬 깊어졌으며, 질의응답 게시판을 통해 다른 동료들의 질문과 정리 방식을 참고하면서 다양한 관점을 접할 수 있었습니다. 이러한 상호적인 학습 경험은 혼자 공부할 때보다 훨씬 효과적이었다고 생각합니다. 수업이 끝난 후 이번 보고서를 작성하면서, 한 학기 동안 배운 통계적 알고리즘의 흐름과 핵심 내용을 다시 한 번 정리해 볼 수 있었습니다. 처음에는 어렵게 느껴졌던 개념들이 보고서를 통해 하나의 흐름으로 연결되었고, 이를 통해 제 학습 과정을 되돌아보는 의미 있는 시간이 되었습니다. 이번 보고서 작성 경험은 한 학기 동안 배운 내용을 다시 돌아보고 정리할 수 있는 좋은 마무리 과정이었다고 생각합니다.

“통계적 알고리즘” PBL 보고서를 작성하며 한 학기 동안의 수업을 마무리하고 돌아보는 시간을 가졌습니다. 이번 수업을 통해 단순히 공식과 알고리즘을 외우거나 무작정 적용하는 것이 아니라, 각 알고리즘이 어떤 수학적 배경에서 출발하고 어떤 상황에서 필요한지 연결하는 과정을 이해하는 경험을 할 수 있었습니다. 특히 도함수, 차분 근사, 수치적분처럼 기초 개념이 왜 중요한지 체감할 수 있었고, 그 기반 위에서 이분법과 뉴턴의 방법, 반복해법, 켤레기울기법 등으로 확장하며 이해할 수 있었습니다. 또한 R 프로그램을 통해 알고리즘을 직접 구현해 보면서 수식으로만 이해한 내용이 실제 계산 과정으로 어떻게 실행되는지 가시적으로 확인할 수 있었고, 그 과정에서 초기값 설정이나 수렴 조건, 오차 기준 같은 요소들이 결과에 얼마나 중요하게 영향을 미치는지 이러한 지점을 왜 고려해야 하는지 알 수 있었습니다. 또한 수업에서 AI 튜터를 활용하며, 막히는 부분을 바로 질문해 답을 구하고, 내가 이해한 내용이 맞는지 바로 점검할 수 있었습니다. 다만, AI의 답변이 의도와 벗어난 답변을 받거나, 부정확한 내용일 경우에는 해당 답변을 주체적으로 판변할 수 있는 능력이 필요함을 느끼기도 하였습니다. 저의 경우에 AI를 통해 구한 답변을 이해하기 어렵거나, 부정확하다고 느낄 시 검색 엔진을 통해 추가적인 정보를 구하거나, 유튜브와 같은 플랫폼을 통해 학습 영상을 시청하기도 하였습니다. 이런 경험을 통해 AI를 무조건으로 의지하는 것이 아닌 하나의 학습 도구로써 활용한다면 개인의 효용성을 극대화할 수 있을 것이라 느낄 수 있었습니다. 마지막으로 이번 수업을 함께한 동료들의 학습활동을 공유하며, 나 자신이 미처 생각하지 못한 개념이나 정보를 확인할 수 있었고, 학습 동기와 방향을 유지하는 데 도움을 받을 수 있었습니다. 정리하며, 이번 “통계적 알고리즘” 수업을 통해 알고리즘을 단순 계산 방법으로 무작정 사용하는 것을 넘어 해당 알고리즘에 적용된 수학적 배경과 통계적 지식을 공부하고 이러한 알고리즘을 사용하기 위한 환경과 조건 그리고 고려가 필요한 지점이 있는지 한 번쯤 생각해 볼 수 있는 역량을 배울 수 있었습니다. 앞으로는 이러한 기본기를 충실히 학습하며, 알고리즘을 사용할 때 근거를 갖춘 해석을 할 수 있도록 노력해야겠다고 생각하였습니다.

PBL 보고서를 작성하면서, 그동안 학습한 내용을 다시 한 번 돌아보고 정리하는 기회를 가질 수 있었습니다. 이 과정에서 내가 배운 알고리즘들을 어떻게 적용했는지, 어떤 부분에서 이해가 부족했는지 점검할 수 있었고, 그로 인해 다시 한 번 중요한 개념들을 명확히 이해하게 되었습니다. 특히, 각 알고리즘의 이론과 코드 구현 방식을 비교하고, 실제 문제에 어떻게 적용할 수 있을지 고민하면서 나만의 방식으로 학습한 내용들을 더 깊이 있게 습득한 것 같습니다. 보고서를 통해 내가 이해한 내용을 정리하고 표현하는 과정 자체가 큰 도움이 되었고, 이 경험을 바탕으로 앞으로 더 나은 학습과 문제 해결 능력을 기를 수 있을 것 같습니다. 또한 동료들의 질의응답을 통해 내가 놓치고 있던 관점들을 발견할 수 있었고, 문제를 다양한 시각에서 바라보는 중요성도 느끼게 되었습니다. 단순히 결과를 도출하는 것에 그치지 않고, 문제 해결 과정 자체를 논리적으로 설명하는 능력이 필요하다는 점을 깨달았습니다. 알고리즘을 선택하는 기준과 시간·공간 복잡도를 고려하는 사고 과정이 점점 익숙해지고 있음을 느낄 수 있었습니다. 앞으로도 이러한 방식의 학습을 지속하며, 보다 복잡한 문제에도 자신 있게 도전해보고 싶습

니다.

한 학기동안 저희가 이해할 수 있도록 친절하게 설명 잘 해주셔서 감사합니다. 강의내용이 생각보다 어렵긴했지만 실습코드와 강의자료를 올려주신 덕분에 제가 나중에 스스로 공부할 때 큰 어려움 없이 공부 할 수 있었습니다. AI튜터를 이용하여 복습하기도 해보고 모르는거에 대해 물어보기도 했습니다. 아직 AI튜터가 완벽하지는 않지만 나름 잘 알려줘서 유용하게 썼습니다. 다시한번 한 학기동안 감사했습니다!!

통계적 알고리즘 PBL 보고서를 마치며, 이번 과제를 통해 다양한 통계적 알고리즘의 개념과 활용 방법을 이해할 수 있었습니다. 특히 경사하강법과 뉴턴의 방법과 같은 최적화 알고리즘을 학습하면서, 이론을 배우는 데서 그치지 않고 이를 실제 문제에 어떻게 적용할 수 있는지 생각해 보는 계기가 되었습니다. 보고서를 작성하는 과정에서 매주 학습한 내용을 정리하며 자신의 학습 과정을 돌아볼 수 있었고, 이를 통해 배운 내용을 체계적으로 정리할 수 있었습니다. 또한 AI 튜터와의 상호작용을 통해 이해가 부족했던 부분을 보완할 수 있었으며, 이러한 과정을 통해 문제를 해결하는 능력이 향상되었다고 느꼈습니다. Q&A 활동을 통해 동료 학생들과 의견을 나누는 과정도 도움이 되었고, 복습 활동을 통해 학습 내용을 다시 한 번 정리할 수 있었습니다. 마지막으로 이번 PBL 보고서는 단순한 과제 제출을 넘어, 통계적 알고리즘을 실제 데이터 분석에 활용하기 위한 기초를 다질 수 있는 의미 있는 경험이었다고 생각합니다. 앞으로도 이번 과제를 통해 얻은 경험을 바탕으로 분석 역량을 더욱 키워 나가고자 합니다.

한 학기를 마무리하면서 지금까지 작성한 Q&A게시판 글과 AI튜터 복습이력을 다시 살펴보았다. 그 과정에서 심프슨 공식, Gauss-Seidel 방법, 콜레기울기법등와 같은 많은 공식들을 배웠다는 것을 느꼈다. 또한 이러한 알고리즘들이 우리 실생활에서도 활용하고 있다는 점을 알게 되었다. 학기 초에는 '이 공식들을 과연 이해할 수 있을까?'라는 의문이 들기도 했지만 이해할 수 있는 공식만큼은 최대한 이해할려고 노력하였고 AI튜터 및 검색엔진을 통해 학습하였다. 이번 학기를 돌아보면서 공식들을 전부 다 이해하지 못했더라도 이해한 공식들이 존재한다는 점에서 뿌듯함을 느꼈다. 앞으로도 지금까지 실습했던 코드와 정리파일은 학기가 끝난 이후에도 계속 참고해야겠다고 생각했다. 또한 방학동안에는 아직 이해하지 못했던 리만합과 같은 공식들을 다시 공부해야겠다는 다짐이 생겼다.

통계적알고리즘 강의를 통해 단순히 공식을 외워 계산하거나 수식을 계산하는 것이 아니라 문제를 수치적으로 해결하는 다양한 알고리즘의 의미를 이해할 수 있었다. 초반에는 개념과 코드를 처음 접하는 것이 많아 어렵게 느껴지만 ai튜터를 활용하며 이해하지 못했던 부분을 하나씩 정리할 수 있었다. 또한 ai튜터한테 질문하는 과제 활동을 통해 스스로 내용을 정리하고 혼자 공부할 때 학습 방식에 대해 큰 도움이 된 것 같다. 그리고 반복 알고리즘과 행렬 기반 방법을 학습하면서 과정도 중요하다는 점을 깨닫게 되었다. 마지막으로 이번 강의를 통해 얻은 알고리즘은 앞으로의 전공 수업과 데이터 분석에도 큰 도움이 될 것이라 생각한다.

통계적 알고리즘 수업을 들으며 가장 많이 느낀 건, '내가 어느 정도 이해하고 있는 부분도 사실은 겉핥기였구나' 하는 점이었다. 수업이 생각보다 어려워서 초반에는 따라가기 벅찼지만, 그래도 하나씩 개념을 뜯어보면서 이해해가는 과정이 꽤 재밌었다. 특히 주성분 분석이랑 요인분석을 비교하는 부분에서 개인적으로 흥미가 많이 생겼는데, 내가 심리학에서 배웠던 지식들이랑 수학적으로 배우는 내용이 연결되는 걸 보는 게 신기했다. 같은 분석이라도 가정이 다르면 완전히 다른 해석이 나온다는 걸 체감했다. 또한, 수업 분위기 자체가 집중도가 높아서 나도 좀 더 진지하게 임하게 된 부분도 있다. 쉬는 시간에도 코드를 다시 열어보거나, 이해 안 되는 부분은 바로 찾아보는 모습들이 인상적이었다. 그런 환경에 있다 보니 자연스럽게 나도 '아 나도 좀 더 해야겠다'는 생각을 하게 됐다. 물론 여전히 부족한 부분도 많고, 수학적 개념들을 다 이해했다고 할 수는 없지만, 그래도 이번 학습을 통해서 최소한 '낯선 개념을 공부할 때 어떻게 접근해야 하는지'는 배운 것 같다. 나처럼 비전공자에게는 이런 경험이 꽤 중요한 것 같다. 개념을 정확하게 이해하려고 노력하다 보면 어느 순간 전체 흐름이 보이는 순간이 있는데, 그게 이번 수업에서 가장 의미

있는 경험이었다. 앞으로 연구나 코드를 짤 때도 이번에 배운 방식대로 차근차근 연결해보려고 한다. 전체적으로 힘들었지만 배운 게 많은 수업이었다.

이번 학기 동안 통계적알고리즘 수업을 따라가면서 처음에는 낯설고 어렵게만 느껴지던 개념들이 점차 하나의 흐름으로 연결되는 경험을 할 수 있었다. 단순히 공식을 외우는 방식이 아니라, 각 알고리즘이 어떤 원리로 작동하는지, 그리고 실제 데이터 문제를 해결하는 과정에서 어떻게 활용되는지 직접 구현하며 체감할 수 있었다. 특히 R 환경에서 알고리즘을 만들어 실행해 본 경험은 그동안 이론으로만 이해하던 내용을 실제 계산 과정과 연결해 주어 큰 도움이 되었다. 또한 반복해법, 최적화 알고리즘, 고윳값 계산 등 다양한 주제들이 서로 어떻게 이어지는지 이해하면서, 데이터 분석 전반의 기반이 되는 사고 구조를 익힐 수 있었다. 처음에는 어려워 보였지만, 과제를 해결하면서 자연스럽게 문제를 쪼개고, 오류를 확인하고, 다시 수정하며 결과를 만들어가는 과정 자체가 학습이었다. AI 튜터와 자료를 활용해 스스로 부족한 부분을 다시 학습하고 보완할 수 있었던 점도 의미 있었다. 무엇보다도, 단순히 “정답을 구하는 기술”이 아니라 “문제를 해결하는 방식”을 배우는 수업이었다고 느낀다. 이번 수업을 통해 얻은 알고리즘적 사고와 구현 경험은 향후 데이터 분석이나 실제 업무, 그리고 일상 속 의사결정 과정에서도 활용할 수 있는 중요한 자산이 될 것 같다. 스스로 성장하는 시간이었고, 어렵지만 배울수록 흥미가 생기는 수업이었다고 정리하고 싶다.

한 학기 동안의 수업 내용을 정리하는 과정은 생각보다 쉽지 않았지만, 그만큼 수업에서 다룬 개념들이 단편적인 지식이 아니라 서로 긴밀히 연결된 사고의 틀이라는 점을 실감할 수 있었다. 이 보고서를 작성하며 수업 내용을 다시 되짚어보는 과정 자체가 복습이자 정리의 시간이 되었고, 그 과정에서 처음에는 잘 이해되지 않았던 개념들도 보다 명확해졌다. 단순히 학점을 위한 강의가 아니라, 사고의 깊이를 넓혀준 수업이었다고 생각이 들었습니다. ■

AI 기반 대학 수업 혁신 사례의 공유와 확산을 위한 아카이브

발간 KELI
홈페이지 http://www.keliedu.org