CS109A: Introduction to Data Science
Lecture 1 - 강의 요약

Pavlos Protopapas, Kevin Rader, Chris Gumb
Harvard University

Fall 2024

Contents

1 강의 개요 (Course Overview)

CS109A는 데이터 과학 입문 과정으로, 통계학과 컴퓨터 과학의 교차점에서 현대 데이터 분석 기법을 다룹니다. 본 강의는 STAT109A, AC209A, CSCIE-109A와 동일한 과정입니다.

1.1 강의 목표 (Course Goals)

이 과정을 통해 학생들은 다음을 배우게 됩니다:

- 데이터로부터 유용한 예측과 인사이트를 도출하는 방법
- 통계적 모델링과 머신러닝 기법의 기초
- 실제 데이터를 다루고 분석하는 실용적 기술
- Python을 활용한 데이터 과학 도구 사용법

1.2 강사진 (Instructors)

Pavlos Protopapas (수석 강사):

- Scientific Program Director, Institute for Applied Computational Science
- 천체물리학 배경, 데이터 과학 및 머신러닝 전문가
- 18년간 하버드에서 데이터 과학 교육

Kevin Rader (공동 강사):

- Senior Preceptor, Statistics Department
- 통계학 박사, 데이터 과학 교육 전문

Chris Gumb (공동 강사):

- Preceptor, IACS
- 소프트웨어 엔지니어링 배경

2 데이터 과학의 역사와 발전 (History and Evolution of Data Science)

2.1 과학적 방법론의 변화 (Evolution of Scientific Methods)

데이터 과학은 다음과 같은 네 가지 패러다임을 거쳐 발전했습니다:

1. 경험적 관찰 (Empirical Observation): 자연 현상의 직접 관찰
2. 이론적 모델 (Theoretical Models): 수학적 방정식을 통한 현상 설명
3. 컴퓨터 시뮬레이션 (Computational Simulation): 복잡한 시스템의 모델링
4. 데이터 과학/머신러닝 (Data Science/ML): 데이터로부터 패턴 발견

2.2 데이터 과학의 응용 분야 (Applications)

현대 데이터 과학은 다양한 분야에서 활용됩니다:

- 의료 진단 (Medical Diagnosis): 질병 예측 및 진단
- 생성형 AI (Generative AI): 텍스트, 이미지, 음성 생성
- 신약 개발 (Drug Discovery): 분자 구조 예측 및 최적화
- 교통 시스템 (Transportation): 자율주행 및 경로 최적화
- 기후 과학 (Climate Science): 기후 변화 예측 및 분석

3 데이터 과학 프로세스 (The Data Science Process)

데이터 과학 프로젝트는 다음 5단계로 구성됩니다:

3.1 1단계: 질문 정의 (Ask an Interesting Question)

핵심 개념: 명확하고 측정 가능한 연구 질문을 설정합니다.
예시:

- 어떤 요인이 주택 가격에 영향을 미치는가?
- 고객 이탈을 예측할 수 있는가?
- 이미지에서 특정 객체를 분류할 수 있는가?

3.2 2단계: 데이터 수집 (Get the Data)

데이터 소스:

- 웹 스크래핑 (Web Scraping)
- API를 통한 데이터 수집
- 공개 데이터셋 (Public Datasets)
- 실험 및 설문조사

도구: Python의 BeautifulSoup, Selenium, pandas 등

3.3 3단계: 데이터 탐색 (Explore the Data)

탐색적 데이터 분석 (Exploratory Data Analysis, EDA):

- 데이터 시각화 (Visualization)
- 기술 통계량 계산 (Descriptive Statistics)
- 이상치 탐지 (Outlier Detection)
- 패턴 및 관계 파악

주요 통계량:
평균 (Mean):

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

분산 (Variance):

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

표준편차 (Standard Deviation):

$$\sigma = \sqrt{\sigma^2}$$

3.4 4단계: 모델링 (Model the Data)

모델 유형:

- 회귀 모델 (Regression Models): 연속적인 값 예측
  - 선형 회귀 (Linear Regression)
  - 다항 회귀 (Polynomial Regression)
  - 정규화 회귀 (Ridge, Lasso)
- 분류 모델 (Classification Models): 범주형 결과 예측
  - 로지스틱 회귀 (Logistic Regression)
  - 의사결정 트리 (Decision Trees)
  - 랜덤 포레스트 (Random Forests)
- 베이지안 모델 (Bayesian Models): 확률적 추론
- 신경망 (Neural Networks): 딥러닝 모델

선형 회귀 예시:
예측 함수:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$$

손실 함수 (평균 제곱 오차):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

3.5 5단계: 결과 전달 (Communicate and Visualize Results)

핵심 요소:

- 명확한 시각화 (Clear Visualizations)
- 결과의 해석 (Interpretation)
- 불확실성 전달 (Uncertainty Communication)
- 실행 가능한 인사이트 (Actionable Insights)

4 강의 구조 및 일정 (Course Structure and Schedule)

4.1 주간 일정 (Weekly Schedule)

주차	주제	내용
1	Introduction	데이터 과학 개요, 데이터 타입, 시각화
2	Data Collection	웹 스크래핑, pandas 기초
3	EDA	탐색적 데이터 분석, 통계량
4	Linear Regression	선형 회귀 모델, 최소제곱법
5	Multiple Regression	다중 회귀, 모델 선택
6	Regularization	Ridge, Lasso, Elastic Net
7	Classification	로지스틱 회귀, 분류 평가 지표
8	Midterm Review	중간고사 준비 및 복습
9	Tree-based Models	의사결정 트리, 랜덤 포레스트
10	Boosting	Gradient Boosting, XGBoost
11	Neural Networks	신경망 기초, 역전파
12	Deep Learning	CNN, RNN 개요
13	Bayesian Modeling	베이지안 추론, MCMC
14	Review & Projects	최종 복습 및 프로젝트 발표

Table 1: CS109A 주간 강의 일정

4.2 강의 형식 (Course Format)

- 강의 (Lectures): 주 2회, 각 90분
- 실습 (Labs): 주 1회, 실습 문제 풀이
- 오피스 아워 (Office Hours): Teaching Fellows와의 질의응답
- 온라인 자료: Ed Discussion, Canvas

항목	비율	세부사항
숙제 (Homework)	30%	5개 과제, 각 6%
퀴즈 (Quizzes)	10%	2개 퀴즈, 각 5%
중간고사 (Midterm)	18%	1회
기말고사 (Final)	22%	1회
프로젝트 (Project)	20%	팀 프로젝트
총계	100%	

Table 2: 성적 평가 구성

5 성적 평가 (Grading)

5.1 평가 구성 (Grade Components)

5.2 숙제 정책 (Homework Policy)

제출 규정:

- 총 5개의 숙제 (각 6%)
- Jupyter Notebook 형식으로 제출
- Late Days: 학기당 3일 사용 가능
- 협업 허용, 단 각자 코드 작성 필수

Late Day 정책:

- 학기당 총 3일의 지각 허용일 제공
- 1일 단위로 사용 (부분 사용 불가)
- 미리 알릴 필요 없음, 자동 적용
- 모두 소진 후에는 하루당 10% 감점

5.3 출석 정책 (Attendance Policy)

중요: 출석은 성적 자격요건입니다!

출석률	최대 성적
90% 이상	A (제한 없음)
80-89%	B+
70-79%	C+
70% 미만	D+

Table 3: 출석률에 따른 성적 상한

출석 인정 방법:

- 강의 참석
- 강의 녹화 시청 (퀴즈 응시)
- 실습 세션 참여

6 프로젝트 (Course Project)

6.1 프로젝트 구조

팀 구성:

- 3-4명으로 구성된 팀
- 자유롭게 주제 선택
- 실제 데이터셋 사용

프로젝트 단계:

1. 제안서 (Proposal): 연구 질문 및 데이터 설명
2. 중간 보고서 (Milestone): 초기 분석 결과
3. 최종 보고서 (Final Report): 완전한 분석 및 결과
4. 발표 (Presentation): 5-10분 팀 발표

6.2 평가 기준

프로젝트는 다음 기준으로 평가됩니다:

- 질문의 명확성: 연구 질문이 구체적이고 측정 가능한가?
- 데이터 분석: EDA가 충분히 수행되었는가?
- 모델링: 적절한 모델을 선택하고 정당화했는가?
- 결과 해석: 결과를 명확하게 전달했는가?
- 코드 품질: 코드가 깔끔하고 재현 가능한가?

7 필수 도구 및 기술 (Required Tools and Skills)

7.1 프로그래밍 언어

Python 3.x (필수)

- NumPy: 수치 계산
- pandas: 데이터 조작
- matplotlib/seaborn: 시각화
- scikit-learn: 머신러닝
- statsmodels: 통계 모델링

7.2 개발 환경

Jupyter Notebook/JupyterLab:

- 대화형 코드 실행
- 시각화 통합
- Markdown 문서화

설치 방법:

```
# Anaconda
conda install jupyter
conda install numpy pandas matplotlib seaborn
conda install scikit-learn statsmodels
```

7.3 버전 관리

Git/GitHub:

- 코드 버전 관리
- 팀 협업
- 포트폴리오 구축

8 데이터 타입 및 시각화 기초 (Data Types and Visualization Basics)

8.1 데이터 타입 분류

8.1.1 양적 데이터 (Quantitative Data)

1. 연속형 (Continuous):

- 정의: 무한한 값을 가질 수 있는 데이터
- 예시: 키, 몸무게, 온도, 시간
- 특징: 소수점 값 가능

2. 이산형 (Discrete):

- 정의: 셀 수 있는 정수 값
- 예시: 학생 수, 판매량, 클릭 수
- 특징: 정수로만 표현

8.1.2 질적 데이터 (Qualitative/Categorical Data)

1. 명목형 (Nominal):

- 정의: 순서가 없는 범주
- 예시: 색상, 성별, 국가
- 특징: 분류만 가능

2. 순서형 (Ordinal):

- 정의: 순서가 있는 범주
- 예시: 학년, 만족도 (낮음/중간/높음), 순위
- 특징: 순서는 있으나 간격은 일정하지 않음

8.2 기술 통계량 (Descriptive Statistics)

8.2.1 중심 경향성 (Measures of Central Tendency)

평균 (Mean):

$$\mu = \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

특징: 모든 값을 고려, 이상치에 민감
중앙값 (Median):

- 정의: 데이터를 정렬했을 때 중간 위치의 값
- 계산: n이 홀수면 $x_{(n+1)/2}$, 짝수면 $\frac{x_{n/2}+x_{n/2+1}}{2}$
- 특징: 이상치에 강건함

최빈값 (Mode):

- 정의: 가장 빈번하게 나타나는 값
- 특징: 범주형 데이터에도 적용 가능

8.2.2 산포도 (Measures of Spread)

분산 (Variance):
모집단 분산:

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$$

표본 분산 (불편 추정량):

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

표준편차 (Standard Deviation):

$$\sigma = \sqrt{\sigma^2}, \quad s = \sqrt{s^2}$$

사분위수 범위 (Interquartile Range, IQR):

$$IQR = Q_3 - Q_1$$

여기서 Q_1은 제1사분위수(25%), Q_3는 제3사분위수(75%)

8.3 시각화 기법 (Visualization Techniques)

8.3.1 단변량 시각화 (Univariate)

1. 히스토그램 (Histogram):
   - 용도: 연속형 데이터의 분포 확인
   - 특징: 구간(bin)으로 나누어 빈도 표시
   - Python 코드:

```
import matplotlib.pyplot as plt
plt.hist(data, bins=30, edgecolor='black')
plt.xlabel('Value')
plt.ylabel('Frequency')
```

2. 박스플롯 (Box Plot):
   - 용도: 사분위수 및 이상치 확인
   - 구성요소:
     - 박스: Q_1부터 Q_3까지 (IQR)
     - 중앙선: 중앙값 (Median)
     - 수염: $Q_1 - 1.5 \times IQR$부터 $Q_3 + 1.5 \times IQR$까지
     - 점: 이상치 (Outliers)

3. 막대 그래프 (Bar Chart):
   - 용도: 범주형 데이터의 빈도 비교
   - 특징: 각 범주별 개수 또는 비율 표시

8.3.2 이변량 시각화 (Bivariate)

1. 산점도 (Scatter Plot):
   - 용도: 두 연속형 변수 간의 관계 파악
   - 패턴: 선형, 비선형, 상관관계 확인

2. 라인 플롯 (Line Plot):
   - 용도: 시간에 따른 변화 추적
   - 특징: 순서가 중요한 데이터

9 Teaching Fellows 및 지원 (Support Staff)

9.1 Teaching Fellows 팀

강의를 지원하는 Teaching Fellows 팀:

- 실습 세션 진행
- 오피스 아워 운영
- 숙제 및 프로젝트 피드백
- Ed Discussion 질문 답변

역할:

1. Head TF: 전체 TF 팀 조율, 강의 지원 총괄
2. Lab TFs: 실습 세션 진행, 실시간 코딩 도움
3. Grading TFs: 과제 채점 및 피드백 제공
4. Discussion TFs: 온라인 질문 답변

9.2 학습 지원 자료

1. Ed Discussion:

- 질문 및 답변 플랫폼
- 24시간 이내 응답 보장
- 학생 간 협력 학습

2. Canvas:

- 강의 자료 업로드
- 과제 제출
- 성적 확인

3. 오피스 아워:

- 주중 매일 운영
- 1:1 또는 소그룹 지도
- 예약 없이 방문 가능

10 자주 묻는 질문 (FAQ)

10.1 강의 관련

Q1: Python 경험이 없어도 수강 가능한가요?
A: 기본적인 프로그래밍 경험이 있다면 가능합니다. 첫 2주간 Python 기초를 다루며, 추가 자료도 제공됩니다.
Q2: 통계학 배경이 필요한가요?
A: 기초 통계(평균, 표준편차)를 알면 유리하지만 필수는 아닙니다. 필요한 통계 개념은 강의에서 다룹니다.
Q3: 프로젝트 주제는 어떻게 정하나요?
A: 본인이 관심 있는 분야의 데이터를 사용하면 됩니다. 예시 주제와 데이터셋 목록을 제공합니다.

10.2 과제 및 평가

Q4: 숙제에서 협업이 가능한가요?
A: 토론은 가능하지만, 코드는 각자 작성해야 합니다. 동일한 코드 제출 시 학사 경고 대상입니다.
Q5: Late Days는 어떻게 사용하나요?
A: 별도 신청 없이 자동으로 적용됩니다. 마감일 이후 제출하면 Late Day가 차감됩니다.
Q6: 출석을 놓치면 어떻게 되나요?
A: 강의 녹화를 시청하고 퀴즈에 응시하면 출석으로 인정됩니다. 단, 출석률에 따라 최종 성적에 상한이 적용됩니다.

10.3 기술적 질문

Q7: 어떤 컴퓨터가 필요한가요?
A: Python과 Jupyter Notebook을 실행할 수 있는 노트북이면 충분합니다. Google Colab(무료)도 사용 가능합니다.
Q8: 데이터셋은 어디서 구하나요?
A: Kaggle, UCI ML Repository, government open data 등을 활용할 수 있습니다. 강의에서 추천 목록을 제공합니다.

11 학습 전략 및 조언 (Study Strategies)

11.1 효과적인 학습 방법

1. 실습 중심 학습:

- 강의를 들은 후 반드시 코드를 직접 작성
- 예제 데이터를 변형하여 실험
- 에러 메시지를 두려워하지 말고 디버깅 연습

2. 점진적 이해:

- 한 번에 모든 것을 이해하려 하지 말 것
- 기본 개념부터 차근차근 쌓아가기
- 이해가 안 되면 질문하기 (Ed Discussion)

3. 프로젝트 기반 학습:

- 관심 있는 분야의 데이터로 연습
- 작은 프로젝트부터 시작하여 확장
- GitHub에 포트폴리오 구축

11.2 시간 관리

주간 학습 계획 (권장):

- 강의 시청: 3-4시간
- 실습 및 복습: 2-3시간
- 숙제: 4-6시간
- 프로젝트: 2-3시간
- 총 주당 약 12-15시간

과제 관리 팁:

- 마감일 최소 3일 전에 시작
- Late Days는 비상용으로 보관
- 막히면 조기에 도움 요청

12 핵심 개념 정리 (Key Concepts Summary)

12.1 데이터 과학의 핵심 원칙

1. 질문 중심 접근 (Question-Driven Approach)
    - 명확한 목표 설정
    - 측정 가능한 성과 지표
2. 데이터 품질 우선 (Data Quality First)
    - Garbage in, garbage out
    - 철저한 데이터 정제 및 검증
3. 반복적 개선 (Iterative Improvement)
    - 프로토타입 → 평가 → 개선

- 지속적인 모델 업데이트

4. 해석 가능성 (Interpretability)
    - 블랙박스 모델 지양
    - 결과의 근거 제시
5. 윤리적 고려 (Ethical Considerations)
    - 편향성 인식 및 완화
    - 개인정보 보호
    - 공정성 및 투명성

12.2 수식 정리 (Formula Reference)

회귀 분석 (Regression Analysis):

단순 선형 회귀:

$$y = \beta_0 + \beta_1 x + \epsilon$$

다중 선형 회귀:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \epsilon$$

정규 방정식 (Normal Equation):

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

모델 평가 지표:

평균 절대 오차 (MAE):

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

평균 제곱근 오차 (RMSE):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$

결정 계수 (R^2):

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

13 다음 강의 예고 (Next Lecture Preview)

13.1 Lecture 2: 데이터 수집 및 처리

다음 강의에서 다룰 내용:

- 웹 스크래핑 (Web Scraping):

  - BeautifulSoup 사용법
  - HTML 파싱
  - 윤리적 스크래핑
- pandas 기초:
  - DataFrame 생성 및 조작
  - 데이터 필터링 및 정렬
  - 결측치 처리
- 데이터 정제 (Data Cleaning):
  - 이상치 탐지 및 처리
  - 데이터 타입 변환
  - 중복 제거

13.2 준비 사항

다음 강의 전에 완료할 것:

1. Python 및 Jupyter Notebook 설치 확인
2. pandas, BeautifulSoup 라이브러리 설치
3. Homework 0 (Python 복습) 완료
4. Ed Discussion에 가입

14 추가 학습 자료 (Additional Resources)

14.1 추천 교재

1. Python for Data Analysis by Wes McKinney
   - pandas 라이브러리 공식 가이드
   - 데이터 조작 및 분석 기법
2. Introduction to Statistical Learning by James, Witten, Hastie, Tibshirani
   - 통계 학습의 기초
   - R 코드 예제 (Python으로 변환 가능)
3. Hands-On Machine Learning by Aur□lien G□ron
   - scikit-learn 활용
   - 실전 머신러닝 프로젝트

14.2 온라인 자료

공식 문서:

- Python: `https://docs.python.org/3/`
- pandas: `https://pandas.pydata.org/docs/`
- scikit-learn: `https://scikit-learn.org/`
- matplotlib: `https://matplotlib.org/`

연습 플랫폼:

- Kaggle: 데이터셋 및 경진대회
- DataCamp: 인터랙티브 Python 학습
- LeetCode: 알고리즘 문제 풀이

15 결론 (Conclusion)

CS109A는 데이터 과학의 전체 파이프라인을 학습하는 종합적인 과정입니다. 이 강의를 통해:

- 실제 데이터를 수집하고 처리하는 능력 배양
- 통계적 모델링 및 머신러닝 기법 습득
- Python을 활용한 데이터 분석 도구 활용
- 결과를 효과적으로 시각화하고 전달하는 방법 학습
- 팀 프로젝트를 통한 실전 경험 축적

성공을 위한 핵심 요소:

1. 꾸준한 실습과 코딩 연습
2. 적극적인 질문 및 토론 참여
3. 프로젝트에 관심 있는 분야 적용
4. 출석 및 과제 마감일 준수
5. Teaching Fellows 및 동료들과의 협력

데이터 과학은 지속적으로 발전하는 분야입니다. 이 강의는 여러분의 데이터 과학 여정의 시작점이 될 것입니다.

행운을 빕니다! Good luck!