December 10, 2025

▣ 강의명: CS109A: 데이터 과학 입문

▣ 주차: Lecture 17

▣ 교수명: Pavlos Protopapas, Kevin Rader, Chris Gumb

▣ 목적: Lecture 17의 핵심 개념 학습

□ 핵심 요약

본 강의는 MCMC (Markov Chain Monte Carlo)와 결측치 처리 방법을 다룹니다. MCMC는 복잡한 확률 분포에서 샘플을 추출하는 강력한 방법이며, 메트로폴리스-헤이스팅스 알고리즘의 원리와 번인(Burn-in), $\hat{R}$ 수렴 진단 방법을 학습합니다. 또한 데이터셋의 결측치를 적절히 처리하는 대치(Imputation) 기법과 그 한계를 이해합니다.

Contents

1 핵심 용어 정리

Table 1: *Lecture 17* 핵심 용어

용어	원어	쉬운 설명	비고 (등장 위치)
MCMC	Markov Chain Monte Carlo	마르코프 체인을 이용해 복잡한 분포에서 샘플을 뽑는 방법.	MCMC
메트로폴리스-헤이스팅스	Metropolis-Hastings	MCMC의 대표적인 알고리즘. "언덕 오르기" 비유.	MCMC
번인 (Burn-in)	Burn-in	MCMC가 안정적인 분포에 도달하기 전까지의 초기 샘플(버려야 함).	MCMC
$\hat{R}$ (R-hat)	R-hat	MCMC가 수렴했는지 판단하는 지표. 1.0에 가까워야 함.	MCMC
결측치	Missing Data	데이터셋에 값이 누락된 상태. (e.g., 'NaN', 'NA')	결측치
대치 (대체)	Imputation	결측치를 추정된 값으로 채우는 것.	결측치

2 MCMC 상세 설명

2.1 R-hat ($\hat{R}$) 통계량

MCMC는 보통 여러 개의 체인(로봇 여러 대)을 동시에 돌립니다. $\hat{R}$은 이 체인들이 모두 같은 산봉우리(안정적인 분포)에 수렴했는지 확인하는 지표입니다.

주의사항

$\hat{R} \approx 1.0$ 이어야 합니다.
만약 $\hat{R} > 1.1$ (혹은 1.05) 이라면, 체인들이 아직 수렴하지 못했거나(e.g., 번인이 부족), 서로 다른 곳(e.g., 어떤 로봇은 A봉우리, 어떤 로봇은 B봉우리)에 가있다는 뜻입니다.

3 학습 체크리스트

☐ 켤레성(Conjugacy)의 의미와, 켤레성이 깨졌을 때(e.g., 로지스틱 회귀) 왜 MCMC가 필요한지 설명할 수 있는가?

☐ MCMC 추적(Trace) 플롯이 안정적인지(수렴했는지) 시각적으로 판단할 수 있는가?

☐ $\hat{R}$ (R-hat) 통계량이 1.0에 가까워야 하는 이유는 무엇인가?

☐ 결측치를 단순히 제거('dropna')하면 안 되는 이유는 무엇인가? (편향 문제)

4 FAQ

Q: $\widehat{\mathrm{R}}$ (R-hat)이 1.0이 아니고 1.3 처럼 나오면 어떻게 해야 하나요?

A: 이는 MCMC가 "수렴에 실패했음"을 알리는 심각한 경고입니다. 여러 대의 로봇(체인)이 서로 다른 산봉우리에 가있거나, 아직 산을 다 오르지도 못했다는 뜻입니다.
해결 방법:

1. 번인(Burn-in) 기간을 늘린다
2. 체인 실행 횟수를 늘린다
3. 더 좋은 초기값(Starting Point)을 설정한다

5 결측치 대치 방법

5.1 단순하지만 나은 방법

- **지시 변수 (Indicator):** 'X_imputed' (0으로 채움) + 'X_was_missing' (1/0)

5.2 가장 좋은 방법

- **확률론적 모델 대치 (Stochastic Imputation)**
- Imputed Value = Model.Predict() + Random Noise(ϵ)

- (분류 문제의 경우: 'predict_proba()' 결과로 **편향된 동전 던지기**)