6

**SK네트웍스 Family AI 과정 14기**

**데이터 전처리 인공지능 데이터 전처리 결과서**



| **산출물 단계** | 데이터 전처리 |
| --- | --- |
| **평가 산출물** | 인공지능 데이터 전처리 결과서(머신러닝 데이터 전처리 결과서) |
| **제출 일자** | 25.09.29 |
| **깃허브 경로** | [https://github.com/skn-ai14-250409/SKN14-Final-3Team-AI](https://github.com/skn-ai14-250409/SKN14-Final-3Team) |
| **작성 팀원** | 이승혁 |

**문서 개요**

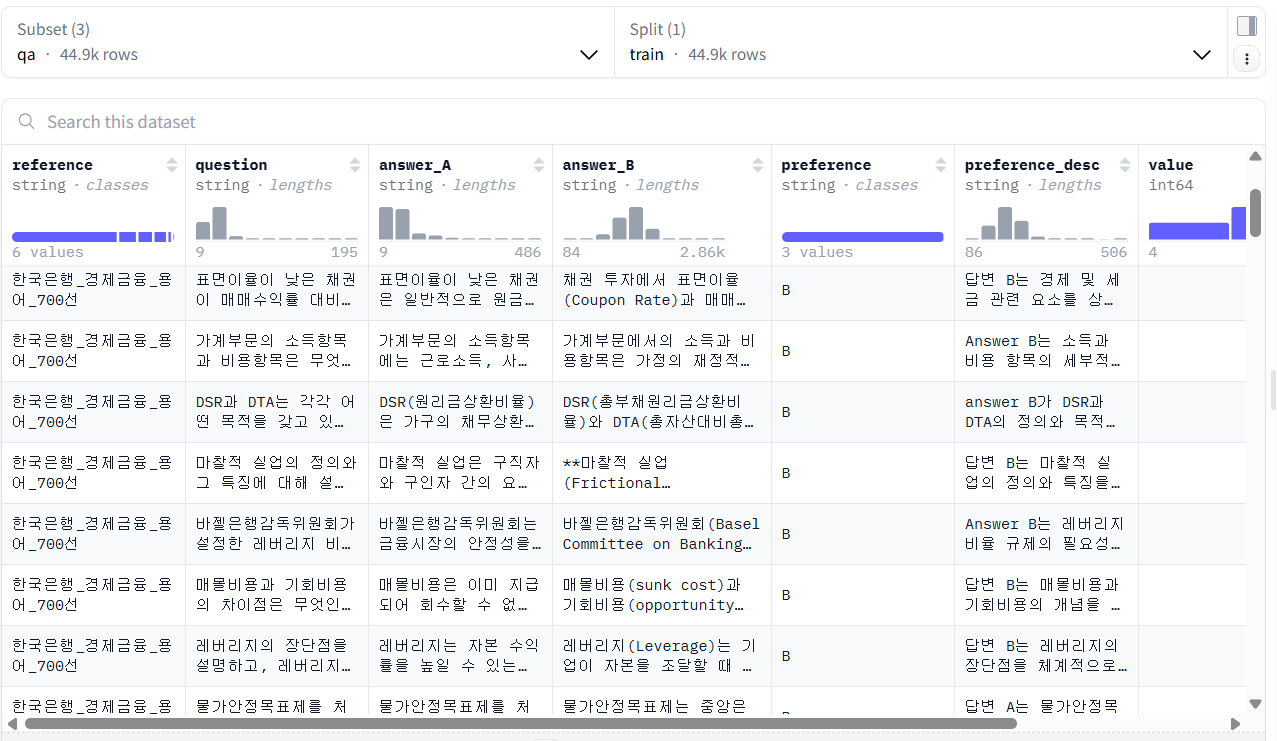
* 프로젝트명: 자체 sLLM을 활용한 은행 여신 업무 지원 AI 플랫폼 개발
* 데이터셋 종류:
  + sLLM을 파인튜닝하기 위한 금융 데이터셋( 금융 데이터 + KB 은행 특화 데이터셋)
  + 여신 심사 모델 머신 러닝을 위한 대출 신청 데이터
* 전처리 목적:
  + 본 팀이 선택한 sLLM 모델, Qwen2.5-7B-Instruct모델에 맞게 데이터를 정제 및 가공
  + 대출 부실 여부를 예측하는 머신러닝 분류 모델을 위한 데이터 정제 및 가공
* 문제 정의 및 목표:
  + Qwen2.5-7B-Instruct 모델을 금융권 데이터와 KB 특화 데이터로 훈련시켜 On-Premise 환경에서 작동할 수 있도록 만든다.
  + 고객(개인/기업)의 정보 및 대출 신청 정보를 바탕으로, 해당 대출이 정상 상환될지(0) 또는 부실 처리될지(1)를 예측하는 이진 분류 모델을 구축한다.

**데이터셋 개요**

* **sLLM 데이터셋 개요**
  + **KB금융그룹의 페르소나를 가진 금융 특화 소형 언어 모델(sLLM)**을 개발하기 위해 제작된 3.1만 건 규모의 대화형 학습 데이터셋
* 전-전처리 단계:
* KB 특화데이터는 KB 홈페이지에서 공개된 KB 국민은행 관련 자료들을 모두 수집
* 총 233개의 pdf 수집
* <https://huggingface.co/datasets/sumilee/SKN14-Final-3Team-Data2> 에 정리
* 해당 pdf 원문 데이터셋을 openai api, gpt 4.1 mini 를 이용.
* 청크사이즈(2,000), 오버랩(200)으로 설정하여 총 1,157개의 Q&A쌍으로 제작
* <https://huggingface.co/datasets/rucipheryn/KB-sLLM-QA-Dataset-Final-Split> 로 업로드
* 데이터 출처 및 수집 방법:
  + 총 31,157건의 데이터를 아래와 같은 소스에서 조합하고 샘플링함.

| 데이터 종류 | 원본 데이터셋 | 사용 건 수 | 내용 |
| --- | --- | --- | --- |
| KB 특화 데이터 | rucipheryn/KB-sLLM-QA-Dataset-Final-Split | 1,157건 (전체) | KB금융 관련 PDF 문서에서 추출한 Q&A |
| 금융 QA 데이터 | aiqwe/FinShibainu (qa 스플릿) | 15,000건 (샘플링) | 일반적인 금융 관련 질의응답 데이터 |
| 금융 MCQA 데이터 | aiqwe/FinShibainu (mcqa 스플릿) | 15,000건 (샘플링) | 객관식 금융 문제 및 정답 데이터 |

* 데이터 전처리 과정
  + aiqwe/FinShibainu의 qa스플릿의 형태:



* + aiqwe/FinShibainu의 mcqa스플릿의 형태:



aiqwe/FinShibainu의 데이터는 qa스플릿과 mcqa 스플릿이 각각 다른 형태로 이루어져 있음

**qa셋은 qustion과 answer\_B를 채택하여 각각 role:user와 role:content로 정의**

근거: **preference\_desc**컬럼에서 나와있다시피 answer\_B의 답변이 더 길고 자세한 것으로 나타남.

mcqa셋은 다지선다 문제로써, question, option, answer의 세 가지 컬럼으로 이루어짐.

**mcqa셋은 qustion과 option을 하나로 묶어 role:user로, answer를 role:content로 정의함.**

근거: sllm의 학습에서 명확한 정답이 있는 SFT방식의 성능이 좋다고 알려져 있음. 실제 테스트 결과도, mcqa셋을 포함한 벤치마크 결과가 높게 나왔음

* 랜덤 샘플링:
  + 각 데이터셋에서 랜덤 샘플링으로 15,000건씩 뽑아내어 셔플함수를 통해 무작위로 추출.
* Qwen2.5 시리즈에 맞는 토크나이저 및 스페셜 토큰을 적용:
  + Qwen2.5 시리즈의 토크나이저 & 스페셜 토큰도 추출하여 적용
* 학습용과 테스트용으로 분할:
  + 데이터 필드: text (string)
  + 분할 정보:
    - train: 29,599건 (95%)
    - test: 1,558건 (5%)
* qa 데이터셋 예시:

<|im\_start|>system You are Qwen, created by Alibaba Cloud. You are a helpful assistant.<|im\_end|> <|im\_start|>user 주식집단의 매도와 동시에 매수하는 선물거래의 의미는 무엇인가요?<|im\_end|> <|im\_start|>assistant 주식집단의 매도와 동시에 매수하는 선물거래는 주식 시장에서의 헤지(hedging) 전략 또는 차익 거래(arbitrage) 전략으로 이해할 수 있습니다... (이하 답변) ...<|im\_end|>

* mcqa 데이터셋 예시:

<|im\_start|>system You are Qwen, created by Alibaba Cloud. You are a helpful assistant.<|im\_end|> <|im\_start|>user T/T 방식의 증가와 신용장 방식의 감소가 시사하는 점은?

A. 은행 간 자금 이동의 신속성이 거래 보안보다 중시된다

B. 기업 간의 신뢰도가 무역 거래의 중요한 요소로 부각된다

C. 무역 거래에서 서류의 역할이 강화되고 있다

D. 금융 기관의 중개 역할이 강화되고 있다<|im\_end|> <|im\_start|>assistant B<|im\_end|>

* 최종 데이터셋
  + https://huggingface.co/datasets/rucipheryn/combined-dataset-30K-final-v4

## 

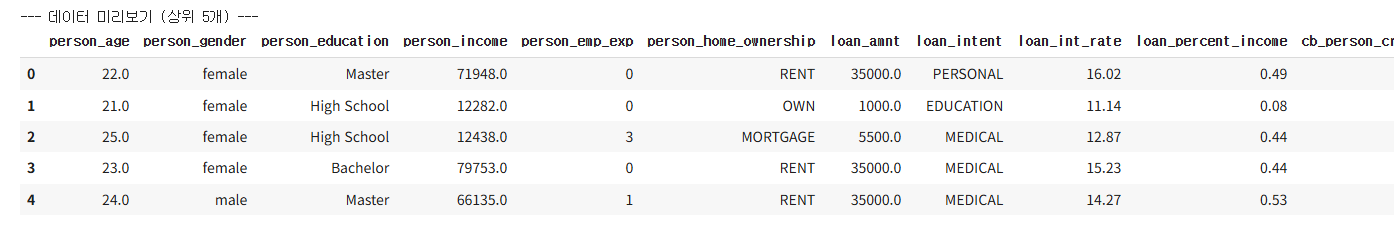
* **머신러닝 데이터셋 개요**
* 데이터 출처 및 수집 방법:
  + 개인 여신 심사 데이터셋   
     [Kaggle - Loan Approval Classification Data](https://www.kaggle.com/datasets/taweilo/loan-approval-classification-data/)에서 csv를 다운로드
  + 기업 여신 심사 데이터셋  
    [Kaggle - US Company Bankruptcy Prediction Dataset](https://www.kaggle.com/datasets/utkarshx27/american-companies-bankruptcy-prediction-dataset)에서 csv를 다운로드

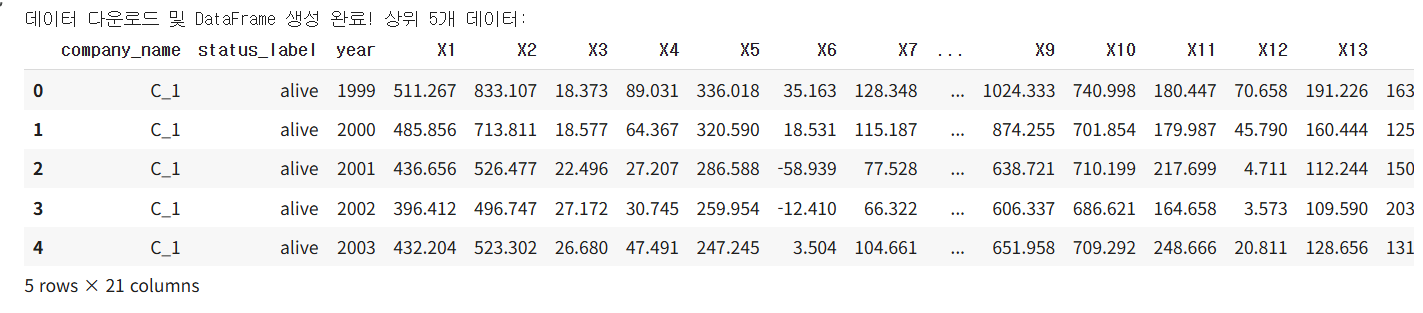
* 데이터 구성(개인):

| **항목명** | **설명** | **예시** |
| --- | --- | --- |
| person\_age | 신청자 나이 | 22 |
| person\_gender | 신청자의 성별 | female |
| person\_education | 신청자의 최종 학력 | Master |
| person\_income | 신청자 연 소득(달러) | 71948.0 |
| person\_emp\_exp | 신청자의 고용 기간 (년) | 3 |
| person\_home\_ownership | 주거 형태 (RENT, OWN 등) | RENT |
| loan\_amnt | 신청한 대출 금액 (달러) | 35000.0 |
| loan\_intent | 대출 목적 (EDUCATION, MEDICAL 등) | PERSONAL |
| loan\_int\_rate | 대출 이자율 | 16.02 |
| loan\_percent\_income | 소득 대비 대출 비율 (loan\_amnt / person\_income) | 0.49 |
| cb\_person\_cred\_hist\_length | 신용 기록 기간 (년) | 3.0 |
| credit\_score | 신청자의 신용 점수 | 561 |
| previous\_loan\_defaults\_on\_file | 과거 대출 연체 기록 | No |
| **loan\_status (Target)** | **대출 상태 (0: 상환 완료, 1: 부실/연체)** | **1** |

* 데이터 구성(기업):

| **항목명** | **설명** | **의미** |
| --- | --- | --- |
| **company\_name** | Company's unique identifier | 회사 고유 식별자 |
| **status\_label** | Company's status (alive or failed) | 회사 상태 (alive: 생존, failed: 파산) - **예측 목표(Target)** |
| **year** | The fiscal year of the data | 데이터의 회계 연도 |
| **X1** | Current assets | 유동 자산 |
| **X2** | Cost of goods sold | 매출 원가 |
| **X3** | Depreciation and amortization | 감가상각비 |
| **X4** | EBITDA | 이자, 세금, 감가상각비 차감 전 이익 |
| **X5** | Inventory | 재고 자산 |
| **X6** | Net Income | 당기 순이익 |
| **X7** | Total Receivables | 총 매출채권 |
| **X8** | Market value | 시장 가치 (시가 총액) |
| **X9** | Net sales | 순매출액 |
| **X10** | Total assets | 총 자산 |
| **X11** | Total Long-term debt | 총 장기 부채 |
| **X12** | EBIT | 이자, 세금 차감 전 이익 |
| **X13** | Gross Profit | 매출 총이익 |
| **X14** | Total Current Liabilities | 총 유동 부채 |
| **X15** | Retained Earnings | 이익 잉여금 |
| **X16** | Total Revenue | 총 수익 (매출) |
| **X17** | Total Liabilities | 총 부채 |
| **X18** | Total Operating Expenses | 총 영업 비용 |

* 원본 데이터 샘플(5~10건 첨부):
  + 개인
  + 기업



**전처리 프로세스 개요**

* 전체 흐름도:
* 개인: ① EDA를 통한 데이터 탐색 → ② 이상치 탐지 및 제거 → ③ 라벨링 오류 수정 → ④ 로그 변환 (정규화) → ⑤ 원-핫 인코딩 (범주형 데이터 변환) → ⑥ 학습/검증 데이터 분리
* 기업: ① EDA를 통한 데이터 탐색 → ② 데이터 구조 변경 (문제 재정의) → ③ 전문 파생 변수 생성 → ④ 다중공선성 기반 변수 선택 → ⑤ 데이터 분할 및 스케일링 → ⑥ 데이터 불균형 처리 (SMOTE)
* 전처리 파이프라인 요약:

| **구분** | **단계** | **목적** | **수행 작업** | **사용 도구/라이브러리** |
| --- | --- | --- | --- | --- |
| **개인** | **이상치 처리** | 모델 성능 저해 요인 제거 | 비정상적인 나이, 경력 데이터 행 제거 | pandas |
| **데이터 변환** | 데이터 의미 정상화 | previous\_loan\_defaults\_on\_file 컬럼의 Yes/No 값을 0/1로 수정 | pandas |
| **정규화** | 데이터 쏠림 현상 완화 | person\_income 컬럼에 로그 변환 적용 | numpy |
| **인코딩** | 문자열 데이터 수치화 | 모든 범주형(object) 데이터를 원-핫 인코딩으로 변환 | pandas |
| **기업** | **라벨링 수정** | 모델의 데이터 누수 방지 및 현실적인 문제 정의 | 파산 기업의 마지막 연도만 'failed'로, 이전 연도는 'alive'로 재정의 | pandas |
| **파생 변수 생성** | 모델 성능 향상을 위한 유의미한 정보 추가 | 부채비율, 유동비율 등 핵심 재무 비율 컬럼 생성 | pandas, numpy |
| **표준화 (스케일링)** | 변수 간 단위 차이로 인한 왜곡 방지 및 학습 안정성 확보 | StandardScaler를 이용해 모든 수치형 데이터 표준화 | scikit-learn |

* 파생 변수 생성 및 의미:
  + 파생 변수 없이 머신 러닝을 적용할 경우, 결과가 아주 좋지 않아 금융권에서 활용하는 파생변수를 자체적으로 생성
  + 파생 변수 없이 머신 러닝 수행 결과:

| **ROC AUC Score** | 0.8762 |
| --- | --- |
| **재현율 (Recall)** | **43%** |
| **정밀도 (Precision)** | 9% |

* 모델의 기본적인 판단 근거를 마련하기 위해, 기업의 **수익성, 유동성, 안정성**을 나타내는 핵심 비율들을 계산하여 추가

| **파생 변수명** | **계산 공식** | **분석 관점** |
| --- | --- | --- |
| **Net Profit Margin** | 당기순이익 / 총수익 | **수익성:** 매출 대비 최종 순이익의 효율성 |
| **Gross Profit Margin** | 매출총이익 / 순매출액 | **수익성:** 제품/서비스 자체의 원가 경쟁력 |
| **ROA (총자산수익률)** | 당기순이익 / 총자산 | **수익성:** 자산 활용 이익 창출 능력 |
| **ROS (매출이익률)** | 당기순이익 / 순매출액 | **수익성:** 판매 및 경영 활동의 종합 효율성 |
| **Current Ratio (유동비율)** | 유동자산 / 유동부채 | **유동성:** 단기 채무 상환 능력 |
| **Quick Ratio (당좌비율)** | (유동자산 - 재고자산) / 유동부채 | **유동성:** 재고를 제외한, 더 보수적인 단기 지급 능력 |
| **Debt to asset ratio** | 총부채 / 총자산 | **안정성:** 자산 대비 부채 의존도 및 재무 위험 수준 |

| **파생 변수명** | **기반 모델** | **핵심 아이디어 및 역할** |
| --- | --- | --- |
| **Altman\_Z** | **Altman Z-Score** | **종합 건강 검진 점수 (높을수록 건전):**  유동성, 안정성, 수익성, 레버리지, 활동성 5개 영역을 종합하여 기업의 총체적인 재무 건전성을 단일 점수로 표현합니다. |
| **Ohlson\_O** | **Ohlson O-Score** | **파산 확률 계산기 (높을수록 위험):**  기업 규모, 부채, 유동성, 수익성 등을 통계적으로 분석하여 파산 확률 자체를 예측합니다. 비선형적 관계를 잘 반영합니다. |
| **Piotroski\_F** | **Piotroski F-Score** | **재무 건전성 채점표 (높을수록 건전):**  수익성, 안정성, 효율성 3개 영역의 9개 항목을 평가하여 기업의 근본적인 체력을 0~9점으로 점수화합니다. 이익의 '질'까지 평가합니다. |

### 파생 변수 생성 및 의미전문 재무 분석 모델 상세 계산식:

각 모델의 스코어는 외부 라이브러리 없이, 데이터셋 내의 재무 변수들을 직접 조합하여 계산

| **모델 / 스코어** | **계산 요소** | **데이터셋 변수를 사용한 계산 공식** |
| --- | --- | --- |
| **Altman Z-Score**  (높을수록 건전) | **A. 유동성** | (X1 - X14) / X10 |
|  | **B. 안정성** | X15 / X10 |
|  | **C. 수익성** | X12 / X10 |
|  | **D. 레버리지** | X8 / X17 |
|  | **E. 활동성** | X9 / X10 |
|  | **최종 Z-Score** | **1.2\*A + 1.4\*B + 3.3\*C + 0.6\*D + 1.0\*E** |
|  |  |  |
| **Ohlson O-Score**  (높을수록 위험) | **A. 규모** | np.log(X10) |
|  | **B. 레버리지** | X17 / X10 |
|  | **C. 유동성** | (X1 - X14) / X10 |
|  | **D. 단기 지급능력** | X14 / X1 |
|  | **E. 수익성** | X6 / X10 |
|  | **최종 O-Score** | **-1.32 - 0.407\*A + 6.03\*B - 1.43\*C + 0.076\*D - 2.37\*E** |
|  |  |  |
| **Piotroski F-Score**  (높을수록 건전) | **A. ROA 수익성** | ( (X6 / X10) > 0 ).astype(int) |
|  | **B. 현금흐름 수익성** | ( X4 > 0 ).astype(int) |
|  | **C. Accrual 품질** | ( X4 > X6 ).astype(int) |
|  | **최종 F-Score** *(단순화 버전)* | **A + B + C** |

**참고:** Piotroski F-Score의 원래 모델은 전년 대비 실적 개선 여부를 평가하는 6개 항목이 더 있으나, 본 프로젝트에서는 데이터 구조의 복잡성을 고려하여 당해 연도 데이터만으로 계산 가능한 3개 핵심 항목(수익성, 현금흐름, 이익의 질)을 기반으로 한 단순화된 버전을 적용

**세부 전처리 단계**

결측치 처리

* 결측치 존재 여부: 없음
* 처리 방법: 초기 데이터 확인(df.info()) 결과, 모든 컬럼에 결측치가 존재하지 않아 별도의 처리를 진행하지 않음.

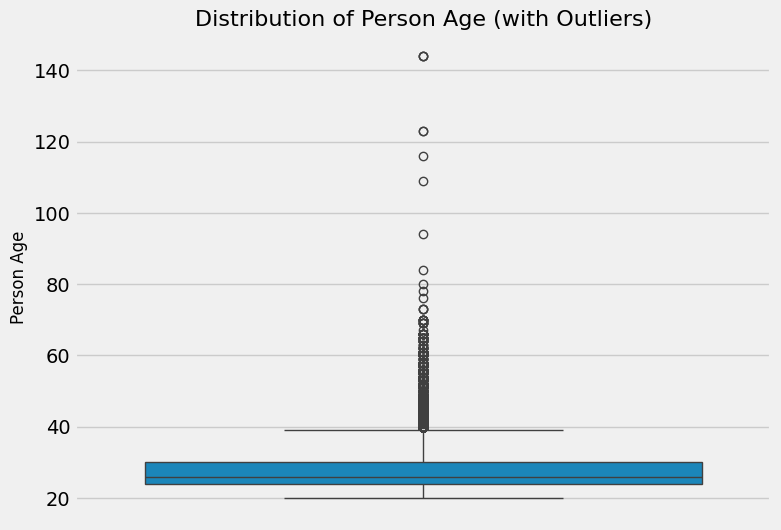
이상치 처리

* 개인:
  + 정의한 이상치 기준:  
     person\_age > 100 또는 person\_emp\_exp > 60 인 경우 (현실적으로 불가능한 값)
* 기업:
  + 대부분의 재무 지표에서 극단적인 값(대기업 데이터)이 다수 발견되었으나, 이는 **파산 예측에 중요한 정보일 수 있어 제거하지 않음.** 대신, 후속 **'표준화(스케일링)'** 단계를 통해 모델 학습에 미치는 영향을 완화함.
* 처리 방식 및 영향:

(개인)

해당 기준을 만족하는 행 전체를 제거함.

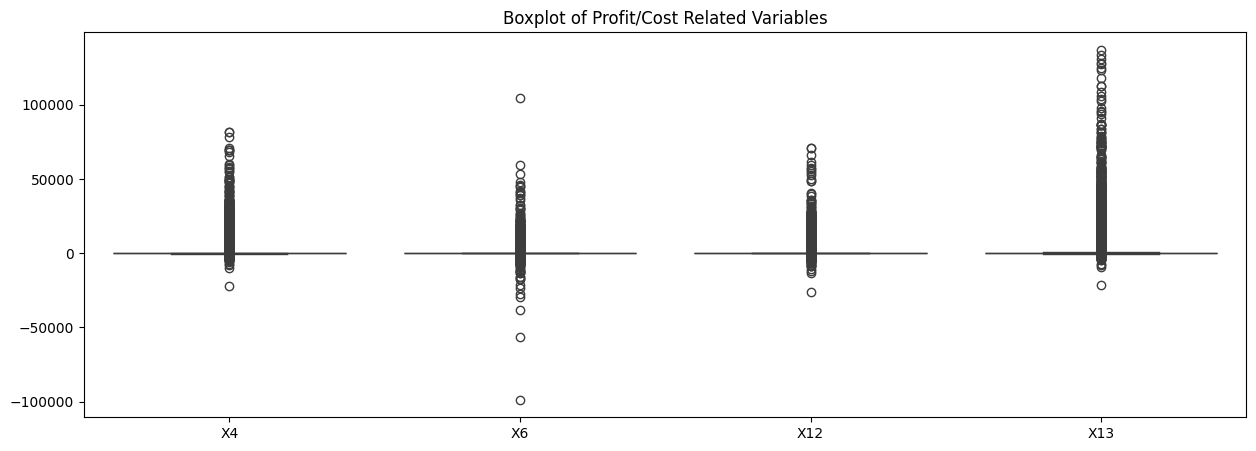
총 **10건**의 데이터가 제거되었으며, 전체 데이터(45,000건) 대비 미미한 수준으로 모델 학습에 영향을 주지 않음.

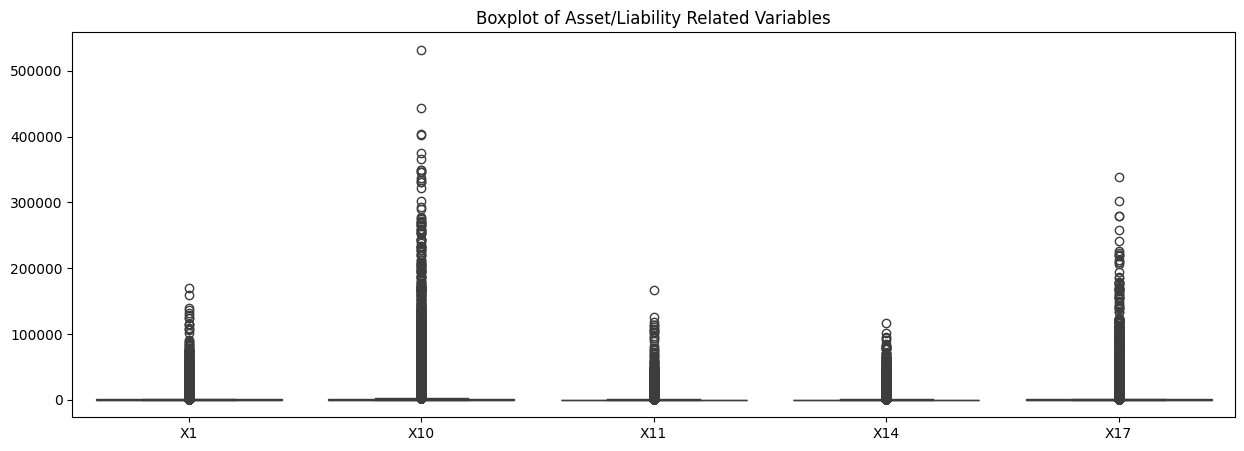


(기업)

기업 데이터의 경우, 이상치를 제거하는 대신 분포를 안정화시키는 방향으로 접근함.

이는 중요한 정보를 보존하면서 모델의 안정성을 높이는 효과를 가져옴.

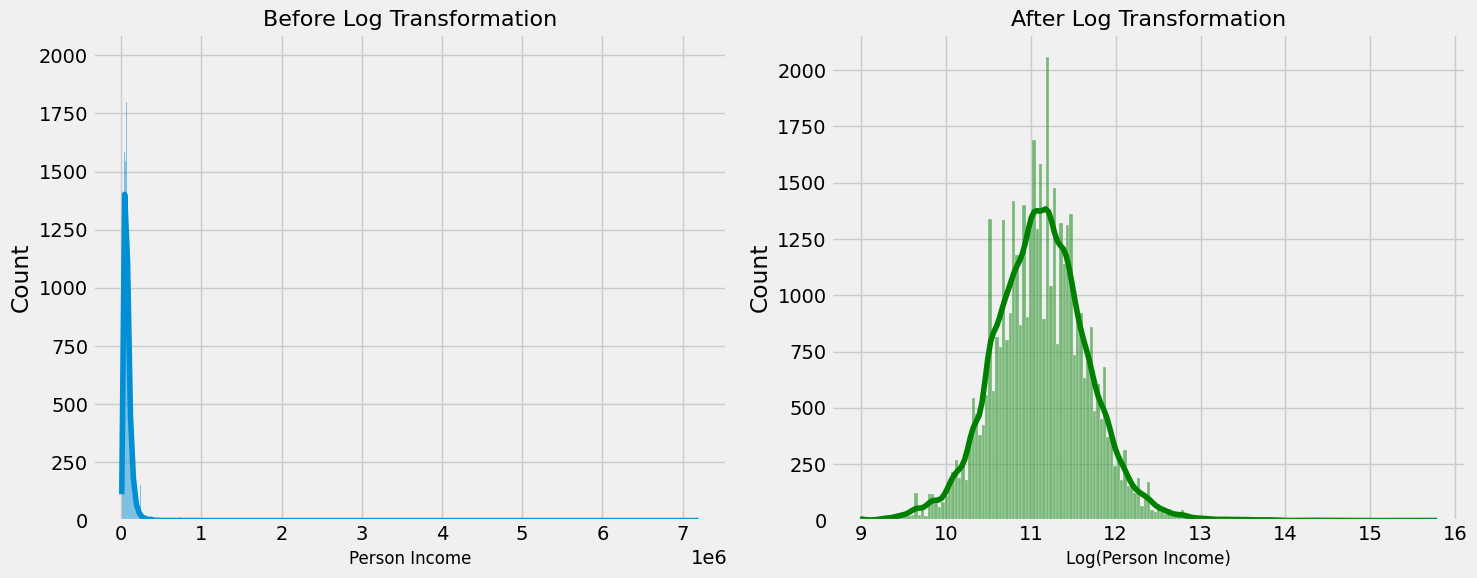




정규화 및 표준화

* 텍스트 정규화:  
  + (개인/기업) 해당 없음 (자연어 데이터가 아님)

(개인)

* 수치형 표준화 (로그 변환):
  + **대상 컬럼**: person\_income
  + **선정 이유**: 극단적인 고소득 값으로 인해 분포가 심하게 왜곡(오른쪽으로 긴 꼬리)되어 있어, 모델의 학습 안정성을 높이기 위해 로그 변환을 적용함.
  + **적용 함수**: numpy.log1p  
      
    df[column] = np.log1p(df[column])

print(f"'{column}' 컬럼에 로그 변환을 성공적으로 적용했습니다.")

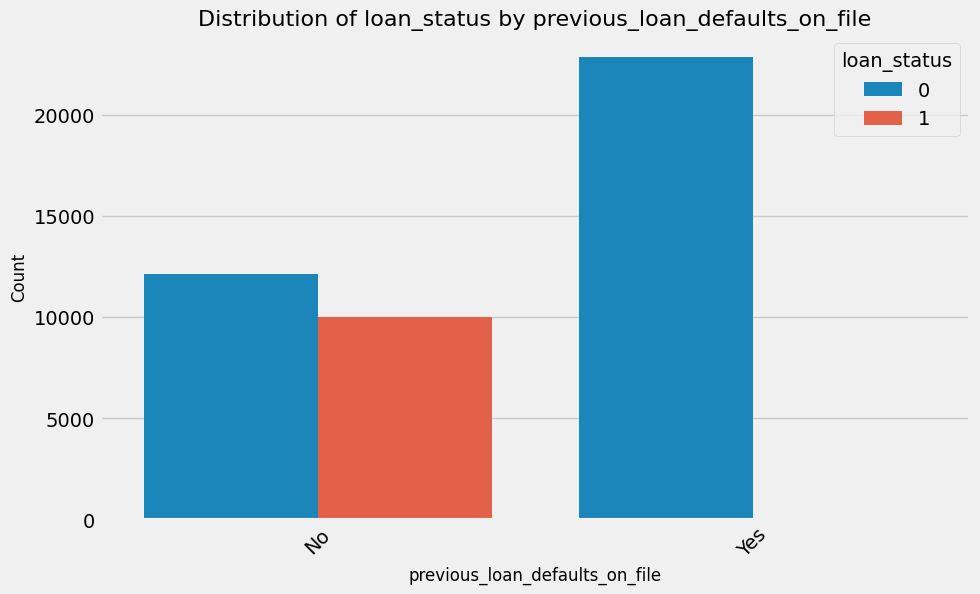
return df

(기업)

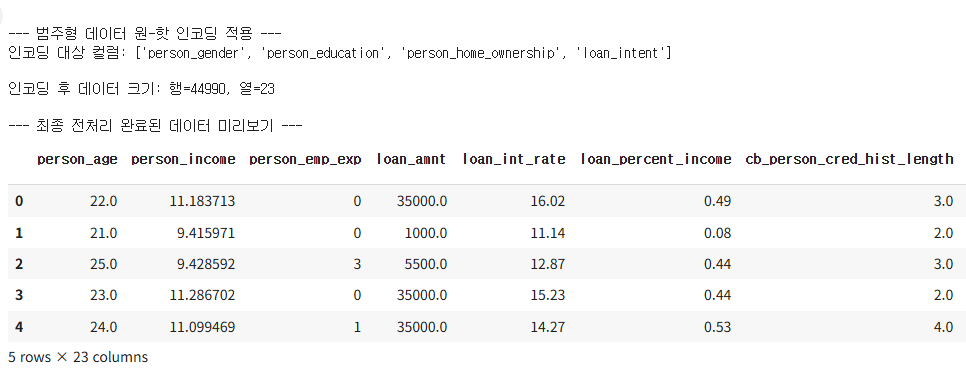
* 수치형 정규화 (로그 변환):
  + **대상 컬럼:** Total assets (Ohlson O-Score 계산 시)
  + **선정 이유:** 기업의 '총자산'과 같은 규모 변수는 극단적인 값으로 인해 분포가 심하게 왜곡되는 경향이 있음. 로그 변환을 통해 데이터의 쏠림 현상을 완화하고, 변수가 모델에 미치는 영향을 안정화시켜 예측 성능을 높임.
  + **적용 방법:** numpy.log() 함수를 Ohlson\_O 스코어 계산 과정에 포함하여 적용함.
* 수치형 표준화 (Standard Scaling):
  + **대상 컬럼:** status\_label, year를 제외한 **모든 수치형 변수** (기본 재무 지표 및 모든 파생 변수 포함)
  + **선정 이유:** 각 재무 지표 및 분석 스코어는 단위(금액, 비율, 점수 등)와 값의 범위가 매우 상이함. StandardScaler를 통해 모든 변수를 **평균 0, 표준편차 1**의 동일한 스케일로 변환하여, 모델이 특정 변수의 크기에 편향되지 않고 공정하게 학습하도록 함.
  + **적용 방법:** sklearn.preprocessing의 StandardScaler를 최종 모델 학습 직전의 모든 Feature에 일괄 적용함.

**데이터 변환 및 생성**

(개인)

* 레이블 인코딩 (의미 수정):
  + **대상 컬럼**: previous\_loan\_defaults\_on\_file
  + **처리 내용**: EDA 결과, 'Yes' 그룹의 부실률이 0%로 나타나는 등 라벨링 오류가 강력히 의심되어, 의미를 바로잡기 위해 값을 변환함 (Yes → 0, No → 1).  
    

* 원-핫 인코딩:
  + **대상 컬럼**: person\_gender, person\_education, person\_home\_ownership, loan\_intent
  + **처리 내용**: 머신러닝 모델이 학습할 수 있도록 모든 문자열 데이터를 수치형으로 변환함. 이 과정에서 전체 컬럼 수가 14개에서 **23개**로 확장됨.



(기업)

* 레이블 인코딩:
  + **대상 컬럼:** status\_label
  + **처리 내용:** 문자열(alive, failed)을 숫자(0, 1)로 변환하여 모델이 학습할 수 있도록 함.
* 데이터 구조 변경 (문제 재정의):
  + **처리 내용:** 시계열 예측 문제의 복잡성을 완화하고, 파산/생존 기업의 공통적인 특성을 효과적으로 학습하기 위해 데이터 구조를 변경함. **'생존' 기업은 중복을 제거**하여 각 기업별 대표 데이터만 남기고, '파산' 기업 데이터와 결합하여 **횡단면(Cross-sectional) 데이터셋을 구축**함.
* 파생 변수 생성: 기본 재무 비율 및 전문 분석 모델 적용
  + **처리 내용:** 모델이 기업의 재무 상태를 다각적이고 깊이 있게 이해할 수 있도록, **기본 재무 비율**과 함께 학계 및 산업계에서 검증된 **전문 재무 분석 모델**의 결과값을 파생 변수로 생성함. (앞서 설명한 표 참조)

**학습/검증 데이터 분리**

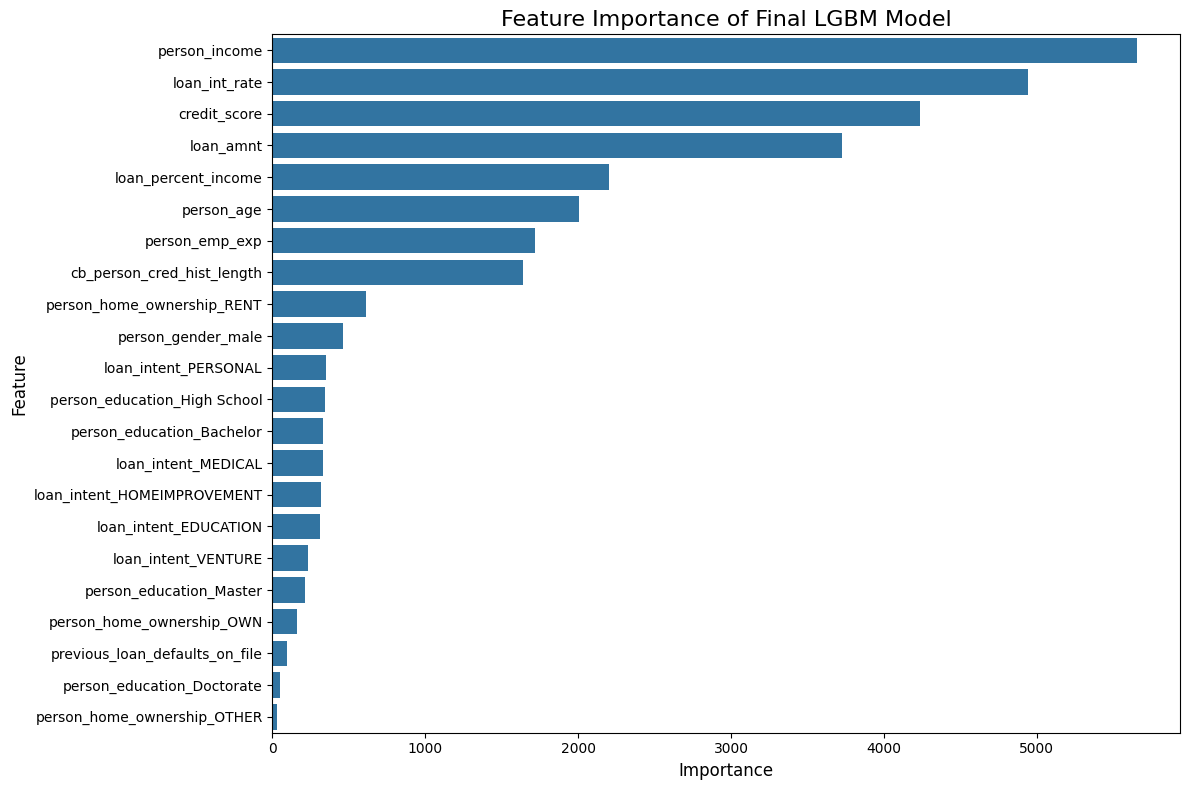
* 분리 기준 및 방법:  
  + (예정) 전처리가 완료된 최종 데이터를 사용하여, 모델 학습을 위한 **학습 데이터(Train set)**와 성능 검증을 위한 **테스트 데이터(Test set)**로 분리.
  + 일반적으로 사용하는 8:2 비율로 무작위 분할(Random Split)을 적용.
  + **stratify 옵션을 사용**하여 학습/테스트 데이터셋 모두에 원본 데이터의 부실(failed) 비율이 동일하게 유지되도록 하여, 평가의 신뢰도를 확보.
* 분리 후 건수:

| 개인/기업 | 구분 | 데이터 수 |
| --- | --- | --- |
| 개인 | 학습 데이터 | 35,992 |
| 테스트 데이터 | 8,998 |
| 기업 | 학습 데이터 | 62,945 |
| 테스트 데이터 | 15,737 |

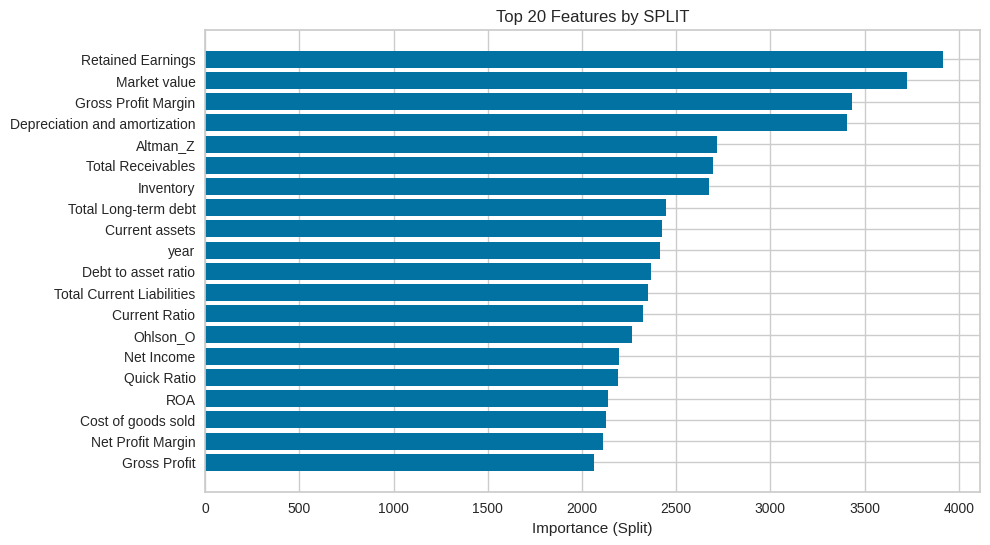
**전처리 결과 요약 및 평가**

* 전처리 후 전체 건수:
  + (개인) 45,000건 → **44,990건** (이상치 10건 제거)
  + (기업) 78,682건 → 13,582 (변수 재생성)
    - **시작 데이터:**
      * df (원본 데이터): **78,682건**
    - **데이터 구조 변경 (중복 제거 및 결합):**
      * failed\_companies: status\_label이 'failed'인 모든 데이터를 필터링.
        + (원본 데이터에서 failed는 5,220건)
      * alive\_companies: status\_label이 'alive'인 데이터에서 company\_name 기준으로 **중복을 제거**.
        + alive 데이터는 총 73,462건이었지만, 실제 회사 수는 8,362개. (df['company\_name'].nunique()로 확인 가능)
      * all\_companies = pd.concat([alive\_companies, failed\_companies], ...): 이 두 데이터를 병합합
    - **최종 건수 계산:**
      * 중복 제거된 '생존' 기업 수: **8,362건**
      * '파산' 기업 수: **5,220건**
      * **합계:** 8,362 + 5,220 = **13,582건**
* 품질 향상 지표:
  + 개인  
    - **이상치 제거**: 논리적 오류가 있는 데이터 10건 정제 완료.
    - **데이터 오류 수정**: 의미가 뒤바뀐 핵심 변수의 라벨링을 정상화하여 데이터 신뢰도 확보.
    - **분포 안정화**: 소득 데이터의 쏠림 현상을 완화하여 모델의 안정성 및 성능 향상 기대.
    - **수치 데이터화**: 모든 데이터를 숫자 형태로 변환하여 머신러닝 모델에 즉시 사용 가능한 상태로 가공 완료.
  + 기업
    - **데이터 오류 수정:** 치명적인 데이터 누수(Data Leakage)를 유발하는 status\_label의 의미를 정상화하여 **데이터의 신뢰도 확보.**
    - **정보 함축성 강화:** 단순 재무 지표를 조합한 **핵심 재무 비율을 추가**하여, 모델이 기업의 건전성을 더 효과적으로 판단할 수 있는 근거를 마련함.
    - **분포 안정화:** StandardScaler를 적용하여 극단적인 값의 영향을 완화하고 모델의 안정성 및 성능 향상을 기대.
    - **수치 데이터화:** 모든 데이터를 숫자 형태로 변환하여 머신러닝 모델에 즉시 사용 가능한 상태로 가공 완료.
* Feature Engineering:

(개인)



(기업)



* 향후 활용 방안:  
  + 이렇게 정제된 데이터는 XGBoost, LightGBM, 로지스틱 회귀 등 다양한 분류 모델의 학습 데이터로 사용하여, 높은 정확도와 안정성을 가진 대출 부실 예측 모델을 구축하는 데 활용될 것이다.

**변경 이력**

| 변경일 | 변경자 | 변경 내용 | 비고 |
| --- | --- | --- | --- |
| 2025-08-23 | 이승혁 | 결과서 최초 작성 | - |
| 2025-08-29 | 이승혁 | 기업 심사 데이터 EDA 및 전처리 프로세스 추가 | - |
| 2025-9-29 | 이승혁 | 최종 sLLM 전처리 및 기업 심사 데이터 EDA 변경 | - |