**최적 모델 선택을 통한**

**default of card clients**

**데이터 결과 분석 및 채무 불이행자 파악**

**7조**

**한양대학교 에리카 산업경영공학과**

**조장** 김동익(2018042588)

**조원**

정영훈(2016006462)

김윤성(2018042624)

정효림(2018042906)

김규현(2018042579)

백재민(2018042679)

**Analysis of default of credit card clients data**

**results and identification of defaulters**

**through optimal model selection**

**group 7**

**HANYANG UNIV. (ERICA) INDUSTRIAL & MANAGEMENT ENGINEERING**

**Leader** Kim dong ik(2018042588)

**Member of a group**

**Jeong yeong hun(2016006462)**

**이름 확인하기 ㅇㅇㅇㅇㅇㅇㅇㅇㅇㅇㅇㅇㅇKim yun seong(201842624)**

**Jeong hyo rim(2018042906)**

**kim gyu hyeon(2018042579)**

**Baek jae min(2018042679)**

**CONTENTS**

1. 서론: 문제 정의
2. 데이터 사전 분석 결과
   1. 분석 내용 소개
   2. 결과 요약
3. 데이터 전처리
4. 모델 제안 (Framework)
5. 제안된 방법 소개
6. 제안한 모델이 앞서 분석한 내용과 어떤 연관이 있는지?
7. 실험 결과
8. 실험 과정 소개
9. 실험 결과 요약 및 원인 분석
10. 결론
11. 결과 요약
12. 프로젝트를 통해 느낀 점 간단히 요약
13. 참고문헌

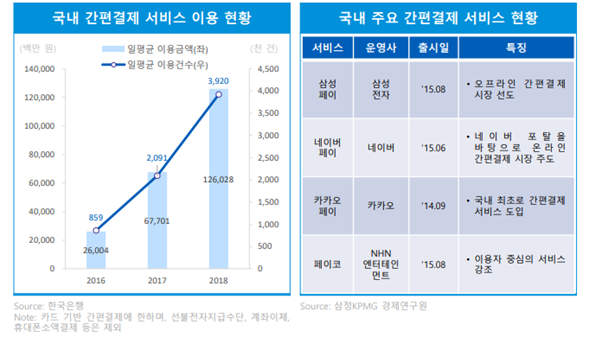
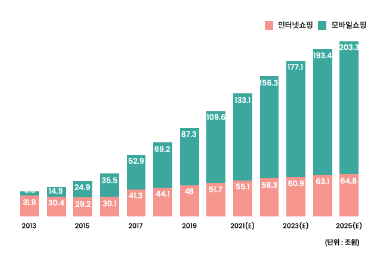
1. 서론: 문제 정의

채무불이행(Default)란, 채무자의 귀책사유로 인해 채무의 내용에 따른 이행이 이루어지지 않는 상태를 통틀어 일컫는 용어이다. 우리나라에서 발생한 채무불이행 관련 사건 사고 중 가장 유명한 것은 흔히 카드대란이라고 하는 2002년에 발생한 가계 신용카드 대출 부실 사태이다. 이 카드대란은 약 수백만 명을 신용불량의 늪에 빠트렸고, 한국 경제와 사회에 큰 타격을 주었다.

그 원인 중 가장 치명적이었던 것은, 신용카드를 발급받아선 안 되는 사람에게까지 신용카드를 발급해 주었던 것에 있다. 당시 정부는 외환위기의 늪을 빠져나오기 위하여 소비를 통한 경기부양과 원활한 세금 징수를 목표로 하였는데, 당시 만연해 있던 지하경제가 그 발목을 잡았다. 이에 정부는 신용카드를 통해 지하경제의 자금을 양지로 끌어올리고자 하였다. 신용카드 사용률을 높이기 위해 정부는 관련 정책, 인프라에 많은 지원을 아끼지 않았고, 기업들도 정부의 정책기조와 발을 맞추어 빠른 속도로 국민들에게 신용카드를 보급했다.

그러나 이 과정에서 카드사의 경쟁 과잉이 심화되었다. 카드사들은 내실을 신경쓰지 않고 오로지 고객 수를 늘리겠다는 맹목적이고 공격적인 방향의 마케팅을 고집했다. 그러나 한국 경제는 외환위기를 완벽히 빠져나온 상태가 아니었고, 경제 구조의 대격변으로 인해 국민들의 소득이 불안정한 상태였다. 또한 외환위기로 생겨난 신규 빈곤층들은 생활비를 자신의 소득이 아니라 현금서비스를 통해 충족하는 일도 왕왕 있었다. 카드사들은 이 점을 간과한 것이다.

신용카드를 발급받아선 안 되는 사람들에게까지 신용카드를 마구잡이로 발급해 주니, 연체율은 점점 늘어만 갔다. 금융에 대한 이해가 충분하지 않았던 신용카드 이용자들 중 일부는 채무에 시달리다 결국 파산하고 말았고, 이러한 사례가 급증하기 시작하며 채권을 회수하지 못한 카드사들은 만성적인 부실에 시달리게 되는데, 이 중 대표격이 바로 현재 신한카드에 인수된 LG카드이다.

그림 1. 글로벌 BNPL 시장 규모 그림 2. 국내 간편결제 시장 규모

20년이 지난 2022년 현재라고 해서 환경이 다르지만은 않다. 최근 IT 공룡들의 공격적인 투자와 경영으로 인해 간편결제 시장이 급격히 성장하고 있는데, 후발주자인 IT 기업들에게 주도권을 내어 주던 기성 카드사들은 이 상황을 타개하기 위해 후불결제(**BNPL**, Buy Now Pay Later) 서비스에 집중하고 있다. 이 후불결제 서비스는 금융 거래 이력이 거의 없는, 소위 씬 파일러(Thin filer)를 대상으로 한다. 씬 파일러에 해당되는 고객들로는 대학생이나 사회초년생 등이 있는데, 이들은 2002년과 똑같이 금융에 대한 이해가 충분치 않은 계층이다. 기존의 신용 평가 모델상 대출을 받을 수 없는 사람들을 대상으로 하는 것이기 때문에 카드사들은 고객의 소비 패턴을 대안정보로 활용하여 분석하는 CSS(Credit Scoring System) 개발에 박차를 가하고 있다.

그 대표적인 예시가 케이뱅크이다. 케이뱅크는 중저신용, 씬 파일러에 특화된 CSS를 개발 완료하여 실제 금융 상품에 적용한 바 있는데, 이 시스템의 가장 큰 특징은 머신 러닝(Machine Learning) 기법을 적극 이용하였다는 점이다. 케이뱅크는 이 시스템을 활용해 중저신용 고객들의 대출 승인율을 18.3%, 씬 파일러 고객들의 대출 승인율을 31.5% 늘렸다고 밝혔다. 또한 고객 수를 늘리는 외적 성장뿐 아니라, 해당 CSS를 보다 고도화하여 리스크 관리 등의 내실 다지기에 역량을 집중하겠다는 포부를 밝히기도 하였다.

<http://www.nspna.com/news/?mode=view&newsid=555541>

<https://zdnet.co.kr/view/?no=20211022101357>

이러한 머신 러닝 기반 CSS에서 가장 중요한 건 역시 채무불이행 예측률이다. 이러한 필요성을 인식하여 국회는 신용정보의 이용 및 보호에 관한 법률을 일부 개정하여, 국무총리실 산하 금융위원회의 감독을 받는 한국신용정보원을 2016년 1월에 설립한 바 있다. 한국신용정보원에서는 채무불이행 예측률 상승을 위하여 2019년 6월 3일 금융빅데이터개방시스템(이하 CreDB)을 구축하였고, 점차 개방 대상을 확대해 나가고 있다.

그림 3. 신용평가사 매출 규모

신용평가사의 매출이 증대되고 있는 점도 해당 분야에 불어닥친 열풍과 무관하지 않을 것이다. 위 그림은 2018년부터 2020년까지의 신용평가사 매출 규모에 대한 그래프인데, 비교 대상인 3개 기업 모두 매출 규모가 꾸준히 우상향하고 있다는 점을 확인할 수 있다.

이러한 정보를 종합해 본 결과, 본 7조는 다음의 잠정적인 결론을 도출하였다.

1. 정부와 기업은 머신 러닝을 활용한 신용평가모델의 고도화를 중요시하고 있음. 그 이유로는 핀테크 기업들의 성장도 있겠지만, 리스크 관리가 가장 큰 이유일 것임.
2. 앞으로 유관 분야의 연구가 지속될 것임.

따라서 본 7조는 보다 적절한 분류 모델의 선정을 위해, 각각의 모델들이 갖는 장단점이 실제 분류 성능에 어떠한 영향을 미치는지 알아보고자 한다.

본 연구는 다음과 같이 구성되어 있다.

제2장: 데이터 사전 분석 결과

제3장: 데이터 전처리 과정

제4장: 모델을 선택하게 된 이유

제5장: 모델 적용 과정 및 모델링 결과에 대한 요약 및 분석

제6장: 최종 결론 및 프로젝트를 진행하며 느낀 점

2. 데이터 사전 분석 결과

A. 분석 내용

**1) 변수 정보**

| **수치형 변수(21개)** | **범주형 변수(3개)** | **클래스(Binary)** |
| --- | --- | --- |
| **ID** | **SEX** | **default payment next month** |
| **LIMIT\_BAL** | **EDUCATION** |  |
| **AGE** | **MARRIAGE** |  |
| **PAY\_0** |  |  |
| **PAY\_2** |  |  |
| **PAY\_3** |  |  |
| **PAY\_4** |  |  |
| **PAY\_5** |  |  |
| **PAY\_6** |  |  |
| **BILL\_AMT1** |  |  |
| **BIL\_AMT2** |  |  |
| **BIL\_AMT3** |  |  |
| **BIL\_AMT4** |  |  |
| **BIL\_AMT5** |  |  |
| **BIL\_AMT6** |  |  |
| **PAY\_AMT1** |  |  |
| **PAY\_AMT2** |  |  |
| **PAY\_AMT3** |  |  |
| **PAY\_AMT4** |  |  |
| **PAY\_AMT5** |  |  |
| **PAY\_AMT6** |  |  |

결측치 없음

Boxplot을 이용해 확인해 본 결과, 이상치가 다수 존재하는 것으로 확인됨.

- 결측치, 이상치 파악을 해본 결과 결측치는 없는 것으로 확인할 수 있었고, 이상치는 모든 변수를 Boxplot을 통해 많이 발견되어 전처리 작업을 진행하게 되었습니다.

그림1. 데이터의 변수 정보

**2) 데이터 분포 확인**

- 채무불이행 값 원그래프로 확인(클래스의 불균형으로 오버샘플링 해야겠다고 말하기, 오버샘플링으로 선택한 이유도 서술) 해당 데이터 세트도 그렇지만 현실의 데이터들은 클래스 불균형이 극심하다. 그 이유는 모델이 적은 수의 클래스의 분포를 학습하지 못함과 모델은 많은 수의 클래스의 분포에 과대 적합되기 때문이다. 따라서 언더샘플링과 오버샘플링 중 하나의 기법을 선택해야 한다.

이 채무 불이행자 판별 모델에서는 채무 불이행자로 예측한 사람 중 실제로 채무 불이행자인 경우를 알 수 있는 정밀도가 잘못 예측되었을 경우보다 실제 채무 불이행자 중 예측한 사람이 진짜 채무 불이행자일 확률인 재현율이 잘못 예측했을 때가 회사에 타격이 더 크다고 판단하였다.

언더샘플링하지 않는 이유는 적은 수의 클래스의 샘플이 매우 적으며, 언더샘플링을 하면 일반적으로 소수 클래스에 대한 재현율 값이 작아지기 때문이다. 하지만 오버샘플링하면 보통 정밀도가 떨어지는 대신 재현율이 오른다. 이 데이터에서는 재현율이 더 중요한 조건이므로 오버샘플링을 진행한다.

학력 변수는 고등 교육을 받을수록 채무 불이행 가능성이 감소하는게 보이고 결혼 여부에 채무 불이행이 미치는 영향이 미세하게 보이기 때문에 결혼 여부가 학력보다 영향을 덜 미치는 변수라는 것을 알 수 있다. 이를 통해, 각각의 변수들에 대한 변수 중요도를 파악할 필요가 있다. 그래서 FEATURE IMPORTANCE를 확인한 결과 최근 3개월 내 변수가 종속 변수에 주는 영향이 3~6개월 변수들보다 크게 나온다는 것을 알 수 있다. 또한, LIMIT\_BAL 분포를 확인했더니 한도 범위(분포 그래프 가로축 0~600000)가 너무 광범위하기에 스케일링(표준화)가 필요하다고 판단했다. 스케일링을 하는 이유는 대부분의 학습 알고리즘은 변수의 데이터 범위가 다를 경우 잘 동작하지 않기 때문이다.

예시로 나이와 재산이라는 변수가 있다면 재산의 값이 나이보다 훨씬 크기 때문에 재산에 치중한 학습을 하게 된다. 이로 인해 스케일링이 필요하다는 것이다. 스케일링 방법에는 0에서 1 사이의 값으로 치환하는 정규화(Normalization), 표준편차 값으로 치환하는 표준화 (Standardization)가 있다. 정규화와 표준화 둘 다 장단점이 있기에 각각 비교해 보는 게 좋다. 또한 정규화는 이상치에 매우 예민하다. 하지만 해당 데이터 세트는 이상을 잡아내기 위한 것이기 때문에 이상치를 제거할 수는 없다. 그렇기에 정규화가 아닌 표준화를 선택하였다. 그리고 데이터 변수 간의 상관관계가 있다면 **다중공선성**이 존재하게 되어 상관관계 분석을 진행해주어야 한다.

* **다중공선성**: 변수 간 강한 상관관계가 존재하면 발생하고 분산이 높아 과적합을 유발하기 때문에 상관관계 분석을 진행해주어야 한다.

B. 결과 요약

데이터 분포를 확인한 결과 클래스의 불균형, 너무 넓은 범위의 데이터를 가진 변수, 변수들의 다른 중요도 가중치, 높은 상관관계를 가지는 변수들을 발견했다. 데이터 전처리 때 오버샘플링, 스케일링, 중요도가 높은 변수들만 남긴 대조군 데이터 셋 생성, 상관관계가 높은 변수들을 편차 평균으로 묶은 파생변수 만들기를 결정했다. 이렇게 모델 예측력을 높여주는데 영향을 주는 요인들을 위 4가지로 설정한 후 설정한 요인들이 실제로 어떤 영향이 있는지 알아보기 위해 실험계획법의 분산 분석을 통해 알아보았습니다.

3. 데이터 전처리

상관관계가 높은 변수 PAY\_AMT 1~6, BILL\_AMT 1~6 변수들을 묶은 새로운 파생변수(1~6까지의 변수들을 편차 평균함, 파생변수를 편차 평균으로 만든 이유 언급)를 추가한 대조군 1 데이터 셋을 만들었다.

소비패턴이 불규칙한 사람들 (돈을 쓰다가 안 쓰는 사람들, 안 쓰는 사람이 갑자기 쓰는 것)은 편차가 크게 나올 것이라고 예상했다. 사용 지출 금액 단위가 개개인 별로 다르기 때문에 똑같은 조건을 만들어주고자 편차를 평균으로 나누어주어 파생변수를 생성해주었다.

최근 3개월 내 변수가 종속 변수에 주는 영향이 3~6개월 변수들보다 크게 나와서 3~6개월 변수들을 뺀 대조군 2 데이터 셋 만들고 대조군 2 데이터에서 편차 평균 파생변수를 만든 대조군 3 데이터를 만들었다. 이렇게 생성된 데이터 원본, 대조군1, 대조군2, 대조군3 데이터 셋들을 스케일링을 한 것과 하지 않은 것으로 나누어 8개의 데이터 셋을 만들어 보았고, 여기에 더해서 이렇게 만들어진 8개의 데이터 셋을 스케일링 한 것과 하지 않은 것으로 나누어 16개의 데이터 셋을 생성했습니다. 또한, 데이터를 TRAIN, TEST SET을 8 대 2 비율로 나누어주었습니다.

4. 모델 제안 (Framework)

1. 제안된 방법 소개

- 나이브 베이즈 분류 모델

데이터가 각 클래스에 속할 특징 확률을 계산하는 조건부 확률 기반의 지도 학습 분류 방법으로서 장점은 간단하고 정확하며 computation cost가 작아서 빠르다. 큰 데이터 셋에 적합하고 연속형보다 이산형 데이터에서 성능이 좋으며 Multiple class 예측을 위해서도 사용할 수 있다.

단점은 feature 간의 독립성이 있어야 한다는 것이다. 하지만 실제 데이터에서 모든 feature가 독립인 경우는 드물다. 많은 장점이 있지만, feature가 서로 상관관계가 없는 독립이어야 한다는 치명적인 단점이 있다.

- 로지스틱 회귀 모형

회귀를 사용하여 데이터가 어떤 범주에 속할 확률을 0에서 1 사이의 값으로 예측하고 그 확률에 따라 가능성이 더 높은 범주에 속하는 것으로 분류해주는 지도 학습으로 장점은 분류 모델일 뿐만 아니라 확률값을 얻을 수 있고 이는 최종적으로 분류만 할 수 있는 모델에 비해 큰 장점이 된다. 또 이진 분류에서 다중 분류로 확장할 수 있다. 단점은 교호 작용을 따로 추가해야 한다는 것과 가중치를 더하는 것이 아니라 곱하는 식이기 때문에 가중치의 해석이 매우 어렵다.

- 랜덤 포레스트 모델

랜덤 포레스트는 decision tree 모델 여러 개를 훈련시킨 후 그 결과를 종합해 예측하는 앙상블 알고리즘이다. 각 decision tree 모델을 훈련시킬 때 **배깅(Bagging)** 방식을 사용한다. 장점은 예측의 변동성이 줄어들며 과적합을 방지한다. 결측치의 비율이 높아져도 높은 정확도를 나타낸다. 하지만 데이터의 수가 많아지면 decision tree보다 속도가 크게 떨어진다.

* **배깅(Bagging)**: 기존 학습 데이터(Original Data)로부터 랜덤하게 '복원추출'하여 만들어진 동일한 사이즈의 데이터 셋을 Bootstrap이라고 하며, 이를 여러 개 만들어 앙상블을 구성하는 여러 모델을 학습시키는 방법이다. 여기서 학습 모델은 정해진 것이 아니며, 어떠한 지도 학습 알고리즘이든 다양하게 활용될 수 있다.

- XGB 모형 소개

Random Forest는 Bagging 방식의 앙상블 모델 중 대표적이라면, XGBoost는 Boosting 방식의 앙상블 모델 중 대표격입니다. Bagging 방식의 모델은 과적합에 유리하다는 장점이 있고, Boosting 방식의 모델은 bias에 유리하다는 장점이 있습니다. XGBoost는 그나마 과적합에 강건하다고 알려지긴 했으나, Bagging 방식의 Random Forest보다 과적합에 강건하지는 않을 것이라는 판단을 했습니다.

- 판단기준 링크 본문 중 "RF are harder to overfit than XGB."

XGBoost는 Gradient Boosting 알고리즘을 분산환경에서도 실행할 수 있도록 구현해놓은 라이브러리이며 여러 개의 Decision Tree를 조합해서 사용하는 Ensemble 알고리즘입니다. 효율성과 유연성이 뛰어나고 과적합을 방지할 수 있으며 신경망에 비해 시각화가 쉽고 직관적입니다. 하지만 학습시간이 느리고 **Hyper Parameter**가 많아 **Hyper Parameter** 튜닝을 하게 되면 더 오래 걸린다는 단점이 있습니다.

* **Hyper Parameter**: 모델링할 때 사용자가 직접 세팅해주는 값을 의미하며, 정해진 최적의 값이 존재하지 않는다. 파라미터와 하이퍼 파라미터를 구분하는 기준은 사용자가 직접 설정하느냐 아니냐이다. 사용자가 직접 설정하면 hyper parameter, 모델 혹은 데이터에 의해 결정되면 parameter이다.

2. 제안한 모델이 앞서 분석한 내용과 어떤 연관이 있는지?

- 나이브 베이즈 모델 - 상관관계가 높은 변수들이 실제로 모델에 얼마나 영향을 끼칠지 알아보기 위해 독립 여부에 성능이 좌우되는 나이브 베이즈 분류기 모델을 선택함

- 로지스틱 회귀 모델 - 쉽고 간단하다는 장점도 있지만, 비선형 데이터에 취약하다는 단점이 있다. 이에 비해 Decision Tree 기반 모델들은 선형 모델들과 달리, 비선형 특징을 갖는 데이터 분석에 용이하다는 차이점이 있다. 위와 같은 장단점을 고려하여 Decision Tree와 비교해 보고자 Logistic Regression을 선택하게 되었다. 로지스틱 회귀의 단점은 고급 알고리즘에 비해 성능이 좋지 않다는 점도 있지만, 무엇보다 비선형 데이터의 분류에 취약하다는 점이 가장 유명합니다. 만약 해당 데이터 세트에서 비선형 데이터가 갖는 비중이 압도적이라면, 비선형 데이터의 분류에 취약하다고 알려진 로지스틱 회귀 모델의 성능지표는 굉장히 낮게 나올 것입니다. 이에 저희는 로지스틱 회귀 모델이, 비선형 데이터 분류에 뛰어나다고 알려진 Tree 기반 모델에 비해 성능지표에서 얼마나 차이가 나는지 확인해 보려 합니다.

- 랜덤 포레스트 모델 - 채무 불이행에 어떤 변수들이 영향을 많이 미치는지 순위를 매기기 위해 랜덤 포레스트 모델을 선택함

- XGB 모델 - 특히 저희 데이터 세트는 이상치를 판단기준으로 삼았기 때문에 이상치를 제거하지 않았고, 때문에 매우 noisy합니다. XGBoost는 데이터 세트가 noisy할수록 과적합이 많이 일어나는 모델이라는 점도 저희의 판단기준 중 하나였습니다. 이상치를 제거하지 않은 저희 데이터 세트에서, XG Boost와 Random Forest 간에 나타나는 성능지표의 차이를 확인해 보면, 앞으로 개인 CB에서 데이터들을 분류할 때 과적합 문제 해결에 얼마나 중점을 둬야 할지 방향을 알 수 있을 것입니다. 또한 XGBoost는 데이터 분석 대회에서 자주 우승하는 모델이기도 합니다. 로지스틱 회귀가 고급 알고리즘에 비해 성능이 좋지 않다는 점을 고려해 보면, 이 또한 구별점이 될 것이라고 생각합니다.

5. 실험 결과

1) 실험 과정 소개

- f1 socre 사용 이유(먼저 모델 성능 평가를 위해 default로 예측한 사람 중 실제로 default인가에 대해 알 수 있는 정밀도와 실제 default 중 예측한 사람이 진짜 default일 확률인 재현율을 조화평균으로 측정해주어 종합적 성능평가가 가능한 f1 score를 1순위로 비교하였고 실제로 채무 이행자한테 불이행했다고 판단하는 경우가 회사에 타격이 더 크다고 판단하여 재현율을 2순위 지표로 잡고 각각 다른 총 8가지 데이터들을 분석하였습니다.) 6개의 데이터 셋을 4개의 모델로 다 confusion matrix를 써서 64개를 만들어 분석함

이때 가장 f1 score 값이 잘 나온 모델을 채택하여 실험계획법을 통해 4개의 요인(스케일링, 오버샘플링, 파생변수 생성, 변수 삭제)들이 진짜 예측력을 향상하는데 도움을 주었는가를 분석

2) 실험 결과 요약 및 원인 분석

- 모델별 원본, 대조1, 대조2, 대조3 데이터를 파악 후(피피티 오버 샘플링별 그래프) F1 스코어에 영향 주는 것 파악한 후 뭐가 왜 영향을 주는지 이유 분석

1. 오버샘플링을 하지 않으면, 실제 0인 것과 실제 1인 것의 비율에 차이가 생기고, 이 차이가 log-odds에 직접적인 영향을 끼치게 되는데, 이 점이 로지스틱 회귀에 있어 치명적인 요인이 된 것으로 예상한다.

*2. 나이브 베이즈*

*3. 랜덤 포레스트*

*4. Xgboost*

*(EX. 로지스틱에서는 오리지널 데이터와 대조군끼리의 차이는 별로 없었지만, 오버 샘플링을 할 때와 하지 않을 때의 점수가 많은 차이가 나는 것을 확인했음, @ 다른 모델도 다 분석)*

*- 각 모델 종류별로 제일 잘 나온 f1 score의 confusion matrix 4개 보여주고 왜 모델별로 제일 잘 나온 데이터의 조건이 그때인지 이유 서술*

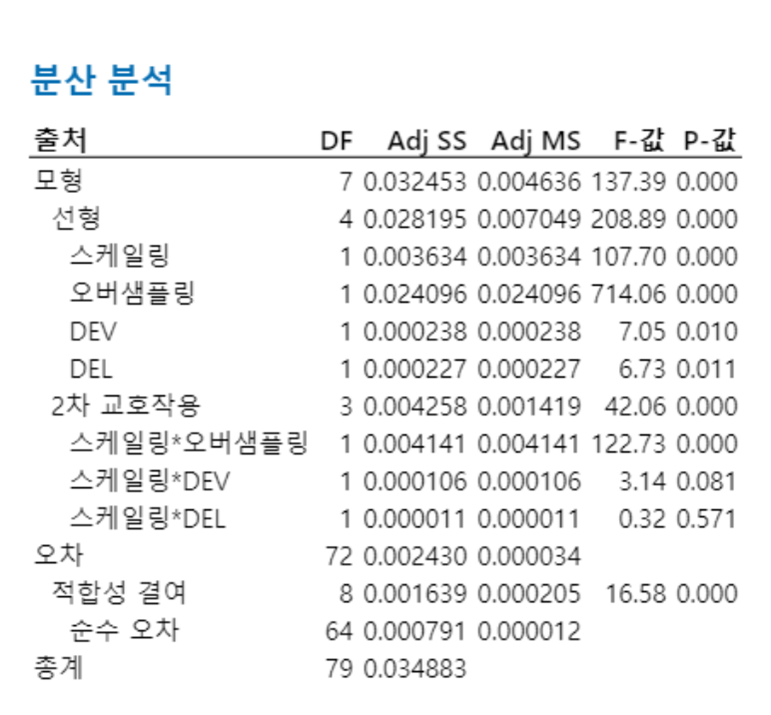
**저는 밑줄 친 내용에 대해선, 그냥 F1 Score 가장 잘 나온 모델 하나, 가장 안 나온 모델 하나만 오버샘플링, 스케일링과 엮어 설명하고 넘어가고 싶습니다. 그리고 이보다 앞서서, 4장에서 언급했었던 것들을 먼저 설명하고 싶어요. F1 Score랑 모델 엮는 건 그 다음에 설명. 일단 그 내용은 다음과 같습니다.**

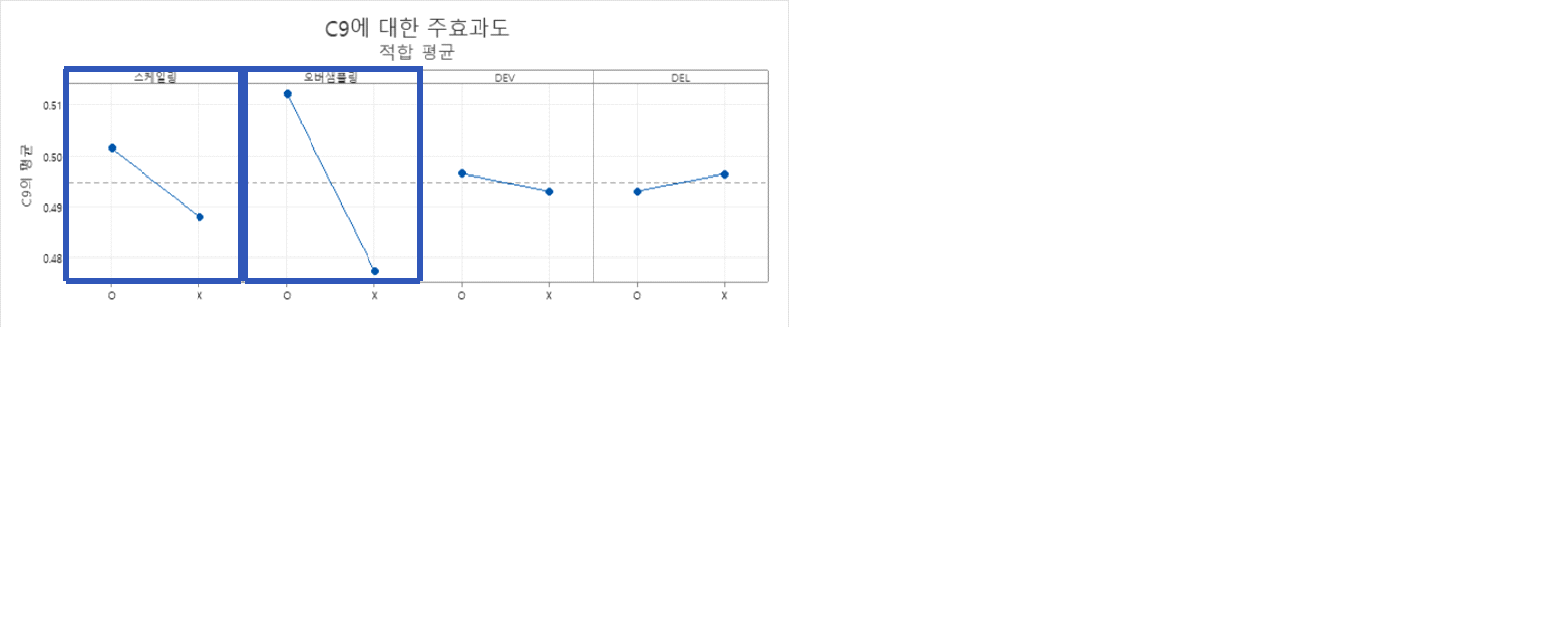
1. 데이터 세트의 비선형성 여부에 따른 로지스틱 회귀 모델의 성능
   1. 로지스틱 회귀에서 나타난 F1-score가 DT 기반 모델들에 비해 크게 뒤떨어지지 않음.
   2. 심지어 성능이 좋기로 유명한 XG Boost와 비교해도 성능이 많이 꿀리지 않음.
   3. 해당 데이터 세트는 선형성을 띄는 것으로 보임. 선형 데이터 분류에 유리한 모델, 혹은 파라미터를 쓰는 게 좋지 않을까? 하는 인사이트.
2. 데이터 세트의 noisy함에 따른 과적합 처리 정도
   1. XGB가 Boosting 기반 모델 치고 아무리 과적합에 비교적 강건하다고 해도, 해당 데이터 세트는 noisy한데, 데이터 세트 특성상 이 noise들을 제거할 수가 없음. 또한 Bagging 기반의 RF가 과적합이 덜 일어난다는 인사이트가 있었음.
   2. 실제로도 RF가 XGB에 비해 F1-Score가 낮게 나왔음. 아마 noise를 제거할 수 없었던 데이터 세트의 특성 때문인 것으로 보임. 즉 데이터 세트 자체가 과적합이 많이 일어날 수밖에 없는 데이터 세트.
   3. 과적합이 일어난 이유들에 대한 고찰이 필요할 것으로 보임. 과적합을 일으키는 요인들을 고려하여 데이터 수집이 이뤄져야 할 것으로 보임.

- 모델 퍼포먼스 꺾은선 그래프로 비교(16개 데이터 셋 중에서 오버샘플링하고 스케일링한 원본 데이터로 비교)

- 제일 f1 score가 잘 나온 랜덤 포레스트 모델을 채택하여 4개의 요인(스케일링, 오버샘플링, 파생변수 생성, 변수 삭제)으로 분산 분석 진행함

- 분산 분석 결과 스케일링과 오버샘플링이 가장 유의미하게 나옴(피피티 그림들 다 첨부)





6. 결론

1. 결과 요약

*모델은 랜덤 포레스트가 가정 f1 score가 가장 높게 나왔으며, 데이터 자료에 스케일링과 오버샘플링은 무조건 해야 하는 것으로 실험계획법을 통해 확인할 수 있었다.*

*랜포를 사용하는 이유가 모델의 노이즈를 심화시키는 과대적합을*

*이상치가 조금 있으니까 그거 때문에 랜덤 포레스트가 좀 더 유의미한 결과?*

1. 로지스틱 선형성 관련
   1. 로지스틱에서 보이는 성능지표, XGB에서 보이는 성능지표를 비교해 보면 그 차이가 크지 않다. 이는 해당 데이터 세트가 비교적 선형성을 띠고 있다고 말할 수 있을 것이다.  
      그러나 해당 데이터 세트의 반응변수들은 대부분 기존 신용평가에서 사용되는 기준들만이 적혀 있었다. 최근 중저신용, 씬 파일러를 대상으로 하는 CSS에서는 기존 금융정보뿐 아니라 대출심사대상자의 관심분야, 소비성향을 신용평가 기준에 넣기도 한다. 심지어 대출심사대상자가 SNS를 하면 페이스북, 인스타그램에 올린 게시물들을 확인해 보기도 하고, 챗봇에서 대출심사를 받을 때 띄어쓰기를 잘하는지까지도 신용평가기준에 넣었다. 하다 못해 KB국민은행에서는 심리 테스트 결과까지 신용평가 판단기준으로 고려하고 있었다.  
      *예를 들면 KB금융지주 경영연구소의 연구보고서, KB지식비타민에서 나온 주장이 있었다. 요는 대출심사대상자를 대상으로 심리 테스트를 진행하고, 그 결과를 신용평가의 요인으로 삼자는 것이다. 그 근거로 든 것이 영국 VisualDNA사에서 진행하는 Big5 기반의 OCEAN이라는 성격평가이다. OCEAN이란 Openness(개방성), Conscientiousness(성실성), Extraversion(개방성), Agreeableness(친화성), Neuroticism(신경성)을 줄인 말인데, 사실 이게 바로 최근 유행하는 MBTI이다. VisualDNA사에선 성실성이 높을수록 신용도가 높더라는 결과를 내놓았다. MBTI에서는 J에 해당되는 이야기이다. 와닿게 이야기하면, P가 아니라 J인 사람들한테 신용평가를 후하게 하자는 말이다.  
      무엇보다 주목해야 할 점은 이게 실제 신용평가에서 사용되고 있었다는 점이다. 이미 영국, 터키, 남아공, 러시아, 폴란드, 슬로바키아의 은행에서 활용 중이었고, 2015년 기준 2년 내로 호주, 체코, 인도, 브라질, 멕시코에도 진출할 예정이라 적혀 있었다.*우리가 여기서 중요하게 봐야 하는 점은, 신용평가 기준이 갈수록 기상천외해지고 있다는 점이다. 이러한 기준들이 반드시 선형성을 띨 것이라 단정할 수는 없다는 점이다. 따라서 우리는 데이터 분석에 앞서 선형성을 먼저 확인해 볼 필요가 있을 것이다.
2. XGBoost 및 과적합 관련
   1. XGBoost는 Outlier에 취약하다.
   2. 과적합을 일으키는 요인은 많다.
      1. 실제 데이터의 일부분을 가져오는 바람에 발생하는 데이터의 편향(표본집단이 모집단을 대표하지 못하는 경우가 있음)
      2. 데이터를 잘못 입력하여 오류가 포함되었을 경우
      3. 과거 학습한 데이터가 지금은 의미가 없을 수 있음.(10년이면 강산도 변한다.)
      4. 변수가 지나치게 많은 경우(차원의 저주)
      5. 모델이 너무 복잡한 경우
   3. 한계점
      1. 실제 데이터(모집단)를 모두 수집하는 것은 불가능
      2. 모든 데이터를 하나하나 학습시키는 것도 불가능
      3. 실제 데이터와의 오차가 발생했더라도, 어디에 오차가 있는지 알 수 없음
   4. XGBoost와 Random Forest와의 비교
      1. 실제로 비교해 보니 XGBoost보다 Random Forest가 성능이 더 좋게 나왔다. 데이터 세트에 Outlier가 너무 많기 때문이기도 할 것이고, 과적합을 일으키는 여러 요인들 때문에 과적합이 일어났기 때문이기도 할 것이다.  
         Outlier의 존재 여부가 신용평가의 핵심임을 고려해 보면, 데이터 전처리 과정에서 무턱대고 Outlier를 제거하는 것은 그릇된 판단인 것으로 보인다. 그렇다면 우리가 고려해야 하는 점은 과적합을 일으키는 요인들에 관한 것이다.  
         우선, 데이터의 편향을 완벽하게 제거한다는 것은 불가능에 가깝다. 그리고 모든 데이터를 하나하나 학습시키는 것도 현재 기술력을 고려하면 어려운 이야기이다. 또한 실제 데이터와 오차가 발생했다 해서 그것이 실제로 어디에서 발생했는지 알아내는 것 역시 어렵다. 과거 학습한 데이터가 현재를 대표하지 못하는 것도 그렇다. 데이터 수집 부서에 데이터 수집을 좀 더 꼼꼼히, 데이터 수집 주기를 좀 더 짧게 해 달라고 요청하는 것이 최선의 방법일 것이다.  
         이 데이터 세트가 과적합이 많이 일어날 수밖에 없는 모델이라면, 우리가 할 수 있는 최선은 주어진 상황에서 과적합 처리에 유리한 방법을 선택하는 것이다. 앙상블 모델이면 부스팅보다는 배깅 기반 모델을 선택하고, 같은 조건이면 좀 더 간단한 설정을 선택하고, 샘플 수가 적으면 안 되니 언더샘플링보다는 오버샘플링을 선호하고, 정규화(Normalization)와 표준화 중 어느 쪽이 과적합이 덜 일어나는지 확인해 보고, 반응변수가 너무 많다 싶으면 규제(Regularization) 방법을 더 고민해 보고, 신경망 모델이면 Dropout을 시도해 보는 식이다.  
         생각해 보니까 과적합이란 게 발생하는 이유는 모델이 너무 세세한 것까지 표현하려 들어서이거나, 데이터가 어떤 부분을 과대대표하는 경우, 그래서 실제 데이터와 예측 데이터와의 오차가 커지는 경우…
   5. 오버샘플링도 결국 과적합 문제
   6. 재현율은?
      1. 정밀도: TP/(TP+FP), 양성이라고 판단했고 실제로도 양성인 비율
      2. 재현율: TP/(TP+FN), 실제 양성 중 모델이 잡아낸 비율
      3. 위양성률: FP/(FP+TN), 양성이 아닌데 양성이라고 판단한 비율
      4. 정밀도와 재현율 중 뭐가 더 중요한지는 사실 알 수 없음. 사업 분야마다 다른 것이니…
      5. 정밀도(or 재현율, 위양성률)를 높이기 위해선 뭘 건드려야 할까? 오버샘플링? 스케일링? 모델은 어떤 거? 하이퍼 파라미터 중에선 어떤 거?
      6. 정밀도(or 재현율, 위양성률)을 높이기 위한 방법이 A라고 하자. 그럼 A를 한 모델에서는 정밀도(or 재현율, 위양성률)가 실제로 올랐나?
      7. 올랐으면 아 효과 있구나, 안 올랐으면… 굳이 설명하지 말고 뭉갭시다

KB지식비타민 연구보고서는 따로 첨부하겠음

https://codedragon.tistory.com/9743

1. 프로젝트를 통해 느낀 점 간단히 요약

7. 참고 문헌

<https://namu.wiki/w/%EC%8B%A0%EC%9A%A9%20%EB%B6%88%EB%9F%89%EC%9E%90>

1. 채무불이행자

[https://namu.wiki/w/LG카드](https://namu.wiki/w/LG%EC%B9%B4%EB%93%9C) 1. 카드대란

[https://namu.wiki/w/단기카드대출](https://namu.wiki/w/%EB%8B%A8%EA%B8%B0%EC%B9%B4%EB%93%9C%EB%8C%80%EC%B6%9C) 1. 현금서비스

[https://m.post.naver.com/viewer/postView.nhn?volumeNo=29940341&memberNo=51956366](https://m.post.naver.com/viewer/postView.nhn%5C?volumeNo=29940341&memberNo=51956366) 그림.1

<https://assets.kpmg/content/dam/kpmg/kr/pdf/2019/kr-issuemonitor-card-new-paradigm-20191104.pdf> 그림. 2

<https://www.kcredit.or.kr:1441/> 1. 한국신용정보원

<https://credb.kcredit.or.kr:3446/frt/main.do> 1. CreDB

[https://www.2e.co.kr/news/articleView.html?idxno=301744](https://www.2e.co.kr/news/articleView.html%5C?idxno=301744) 그림. 3

[https://www.wikileaks-kr.org/news/articleView.html?idxno=125413](https://www.wikileaks-kr.org/news/articleView.html%5C?idxno=125413) 1. BNPL 기사

<https://www.niceinfo.co.kr/creditrating/cb_score_1_4_1.nice> 1. NICE 신용평가정보

<https://bkshin.tistory.com/entry/%EB%A8%B8%EC%8B%A0%EB%9F%AC%EB%8B%9D-1%EB%82%98%EC%9D%B4%EB%B8%8C-%EB%B2%A0%EC%9D%B4%EC%A6%88-%EB%B6%84%EB%A5%98-Naive-Bayes-Classification> 4-1 나이브베이즈

<https://ark-hive.tistory.com/82> 4- 1. 비선형분류모형

<https://tpwkcorqhd.tistory.com/48> 4-1. 로지스틱 회귀분석

<https://hleecaster.com/ml-logistic-regression-concept/> 4-1. 로지스틱 회귀분석

<https://heytech.tistory.com/149> 4-1. 랜덤 포레스트

<https://ybeaning.tistory.com/17> 4-1. 앙상블 모형/ Bagging 과 boosting

[https://bkshin.tistory.com/entry/머신러닝-13-파라미터Parameter와-하이퍼-파라미터Hyper-parameter](https://bkshin.tistory.com/entry/%EB%A8%B8%EC%8B%A0%EB%9F%AC%EB%8B%9D-13-%ED%8C%8C%EB%9D%BC%EB%AF%B8%ED%84%B0Parameter%EC%99%80-%ED%95%98%EC%9D%B4%ED%8D%BC-%ED%8C%8C%EB%9D%BC%EB%AF%B8%ED%84%B0Hyper-parameter) 4-1. Hyper parameter