Feature Engineering 보고서

포하둥우

공영경 김정하 홍재성

Ⅰ. submission_1

1. 결측치 제거

Columns	fill value	기타
반려동물	"정보없음"	
우선청소		
매니저사용휴대폰		
매니저이동방법		
부재중여부	0	
CS교육이수여부		
청소교육이수여부		
결재형태	신용카드	train 결재형태의 최빈값
평수	2	{ '10평대' : 0, '20평대' : 1, '30평대' : 2, '40평대이상' : 3 } 로 mapping 후 2로 채움
고객가입일	test['최초서비스일'] - (train['서비스일자'] - train['고객가입일']).mean()	

2. Feature generation

1) 일자 관련 columns date time으로 변경

- '접수일','접수시각', '최초서비스일', '서비스일자', '서비스시작시간', '서비스종료시간', '고객가입일'

2) 서비스요일

- 서비스일자에 dt.weekday( )를 적용하여 요일로 변경

3) 고객가입일 관련 피쳐

- 고객가입일을 year , month , day, weekday로 나눠줌

4) 지역_매칭

- 서비스주소와 근무가능주소가 동일한 경우 1 , 아닌 경우 0

5) 총서비스시간

- 서비스종료시간 – 서비스시작시간

6) 서비스월

- 서비스일자의 month

7) 서비스-접수-차이

- 서비스일자와 접수일과의 차이

8) 매니저연령 , 매니저연령대 , 매니저연령_qcut

- 매니저생년월일을 활용하여 매니저연령 , 매니저연령대를 계산

- train데이터의 매니저연령 분포를 토대로 임의로 분위수를 구하여 0~4를 할당

9) 부재중_일치

- 부재중서비스가능여부 , 부재중여부를 문자로 변환하여 결합

10) 반려동물여부

- 반려동물이 "없음", "정보없음"에 해당하는 경우 0 , 아닌 경우 1

11) 서비스이용기간

- 서비스일자 - 최초서비스일

12) 회차별일자

- 서비스이용기간 / 현재회차

13) 가입_최초서비스

- 최초서비스일 - 고객가입일

14) 접수시간

- 접수시각 column에 결측값을 -1로 채움

15) 서비스진행비율

- 현재회차 / 전체회차

16) 서비스연도

- 서비스일자의 연도

17) 서비스계절

- 서비스일자의 month가 12,1,2인 경우 겨울 , 3,4,5인 경우 봄, 6,7,8인 경우 여름, 9,10,11인 경우 가을로 나타냄

18) 이용연도

- 고객가입일의 연도만 뽑아 - 2021 + 1

19) 지역

- 서비스주소를 공백을 기준으로 split해서 가장 첫번째 값을 enumerate후 지역에 map

20) 서비스시작시각

- 서비스시작시간을 6으로 나눈 몫이 0인 경우 1로 아닌 경우 0

21) 서비스진행비율_qcut

- 서비스진행비율을 0.2 , 0.4 , 0.6, 0.8, 1로 각 구간에 맞게 값 할당

22) 서비스진행비율_qcut

- train 데이터의 서비스진행비율 분포를 토대로 임의로 분위수를 구하여 0~4를 할당

23) 서비스_접수_차이_qcut

- train 데이터의 서비스-접수-차이 분포를 토대로 임의로 분위수를 구하여 0~4를 할당

24) 서비스이용기간_qcut

- train 데이터의 서비스이용기간 분포를 토대로 임의로 분위수를 구하여 0~2를 할당

25) 접수시간_qcut

- train 데이터의 접수시간 분포를 토대로 임의로 분위수를 구하여 0~3을 할당

26) 가입_최초서비스_qcut

- train 데이터의 가입_최초서비스 분포를 토대로 임의로 분위수를 구하여 0~4를 할당

27) 회차별일자_qcut

- train 데이터의 회차별일자 분포를 토대로 임의로 분위수를 구하여 0~4를 할당

28) 전체회차

- 전체회차가 10 이상인 경우 12로 변경

29) 매니저_교육이수정도

- CS교육이수여부 + 청소교육이수여부

30) cus_type

- 기존고객여부 , 결재형태 , 평수 , 반려동물여부 , 주거형태, 서비스주소를 결합

(train과 test의 교집합이 아닌 경우 기타로 대체)

31) cus_type_count_1

- train 데이터에서 매칭성공여부가 1인 경우, cus_type별 횟수를 count하여 mapping

(결측값은 0으로 채워줌)

32) house_type

- 평수, 반려동물여부, 주거형태, 서비스주소를 결합

(train과 test의 교집합이 아닌 경우 기타로 대체)

33) house_type_count_1

- train 데이터에서 매칭성공여부가 1인 경우, house_type별 횟수를 count하여 mapping

(결측값은 0으로 채워줌)

34) manager_type

- 매니저사용휴대폰, 매니저이동방법, 근무가능지역, CS교육이수여부, 청소교육이수여부 , 추천인여부를 결합 (train과 test의 교집합이 아닌 경우 기타로 대체)

35) manager_type_count_1

- train 데이터에서 매칭성공여부가 1인 경우,manager_type별 횟수를 count하여 mapping

(결측값은 0으로 채워줌)

36) ~별count

- 아래의 columns에 대하여 train 데이터에 매칭성공여부가 1인 횟수를 count 후 mapping

('결재형태','우선청소','매니저사용휴대폰','서비스계절','부재중_일치' ,'평수' , '부재중서비스가능여부' , '추천인여부' , '서비스요일' , '총서비스시간' , '매니저연령대' , '매니저연령_qcut' , '반려동물여부' , '서비스이용기간_qcut' , '서비스진행비율_qcut' , '매니저_교육이수정도' , '이용연도' , '주거형태','우선청소','매니저이동방법', '서비스이용기간_qcut' ,'회차별일자_qcut')

3. Encoding & Scaling

방식	적용 columns	기타
LabelEncoder	'결재형태','주거형태','우선청소','매니저사용휴대폰','매니저이동방법','서비스계절','부재중_일치', 'house_type','manager_type' ,'cus_type'	
Mean Encoding	'결재형태','주거형태','우선청소','매니저사용휴대폰','매니저이동방법','서비스계절','부재중_일치' , '평수' , '부재중서비스가능여부' , '추천인여부' , '서비스요일' , '총서비스시간', '매니저연령대' , '매니저연령_qcut' , 'house_type' , 'manager_type' , 'cus_type' , '반려동물여부' , '서비스이용기간_qcut' , '서비스진행비율_qcut' , '매니저_교육이수정도' , '이용연도' , '주거형태','우선청소','매니저이동방법', '서비스이용기간_qcut' ,'회차별일자_qcut'	- Encoding한 값을 새로운 열로 만들어줌 - test데이터의 결측값은 train 데이터의 평균으로 채워줌
StandardScaler	'고객가입_year','manager_type_count_1' , 'house_type_count_1' , '결재형태별count', '주거형태별count', '우선청소별count', '매니저사용휴대폰별count', '매니저이동방법별count', '서비스계절별count', '부재중_일치별count', '평수별count', '부재중서비스가능여부별count', '추천인여부별count', '서비스요일별count', '총서비스시간별count', '매니저연령대별count', '매니저연령_qcut별count', '반려동물여부별count', '서비스이용기간_qcut별count', '서비스진행비율_qcut별count', '매니저_교육이수정도별count', '이용연도별count', '회차별일자_qcut별count'	

4. 불필요한 column 제거

'접수시각','접수일','최초서비스일','서비스일자','서비스주소','고객가입일','매니저생년월일','매니저성별','근무가능지역','반려동물' ,'house_type'

5. Feature Selection

XGBClassifier를 활용하여 SelectPercentile로 Feature Selection을 진행함

5. ADASYN

ADASYN를 사용하여 오버샘플링을 진행함

Ⅱ. submission_2

1. 결측치 제거

Columns	fill value	기타
반려동물	"정보없음"	
우선청소		
매니저사용휴대폰		
매니저이동방법		
평수		
부재중여부	0	
CS교육이수여부		
청소교육이수여부		
결재형태	신용카드	train 결재형태의 최빈값
고객가입일	test['최초서비스일'] - (train['서비스일자'] - train['고객가입일']).mean()	

2. Feature generation

1) ~ 28) submission_1의 feature와 같음

29) 서비스시작시간

30) 서비스종료시간

31) 교육 여부

- 값이 1이면 이수, 0이면 미이수, 3이면 정보없음으로 값 변환

- CS교육이수여부와 청소교육이수여부를 결합

32) 가입_접수_차이

- 고객가입일과 접수일의 차이

33) 접수주기

- 가입_접수_차이 / 전체회차

34) 매니저가입이후년차

- 고객 접수 시 해당 매니저의 년차를 계산

35) 서비스분기

- 서비스월이 1,2,3월이면 1분기, 4,5,6이면 2분기, 7,8,9 면 3분기 10,11,12이면 4분기로 변환

36) 서비스시작_하루

- 서비스시작시간이 1~6이면 새벽_시작, 7~12이면 아침_시작, 13~18이면 점심_시작, 나머지는 밤_시작으로 변환

37) 서비스시작_종료

- 서비스시작시간이 1~6이면 새벽_종료, 7~12이면 아침_종료, 13~18이면 점심_종료, 나머지는 밤_종료로 변환

38) time_type

- 서비스계절, 서비스요일, 서비스시작_하루, 서비스종료_하루를 결합

(train과 test의 교집합이 아닌 경우 기타로 대체)

39) house_type

- 평수, 반려동물여부, 주거형태. 서비스주소를 결합

(train과 test의 교집합이 아닌 경우 기타로 대체)

40) manager_type

- 매니저사용휴대폰, 매니저이동방법, 근무가능지역, CS교육이수여부, 청소교육이수여부, 추천인여부를 결합

(train과 test의 교집합이 아닌 경우 기타로 대체)

41) customer_type

- 장기서비스여부, 기존고객여부, 서비스주소, 쿠폰사용여부, 우선청소, 부재중여부를 결합

(train과 test의 교집합이 아닌 경우 기타로 대체)

42) region_type

- 서비스주소, 근무가능지역, 매니저이동방법을 결합

(train과 test의 교집합이 아닌 경우 기타로 대체)

43) kmeans_manager

- 매니저사용휴대폰, 매니저이동방법, 근무가능지역, CS교육이수여부, 청소교육이수여부, 부재중서비스가능여부, 추천인여부, 매니저연령_qcut를 통해 매니저 군집 분석 진행

- 총 7개의 매니저 군집 생성

44) kmeans_customer

- 장기서비스여부, 기존고객여부, 결재형태, 서비스주소, 쿠폰사용여부, 우선청소, 부재중여부를 통해 고객 군집 분석 진행

- 총 7개의 고객 군집 생성

36) ~별count

- 아래의 columns에 대하여 train 데이터에 매칭성공여부가 1인 횟수를 count 후 mapping

('장기서비스여부', '기존고객여부', '결재형태', '주거형태', '평수', '부재중여부', '우선청소', '쿠폰사용여부', '매니저사용휴대폰', '매니저이동방법','CS교육이수여부', '청소교육이수여부', '부재중서비스가능여부', '추천인여부', '서비스요일','서비스월', '매니저연령대', '부재중_일치', '반려동물여부', '접수시간', '서비스연도', '서비스계절','교육여부','이용연도', '지역','서비스분기','house_type', '매니저연령_qcut', '서비스이용기간_qcut', '서비스진행비율_qcut', '회차별일자_qcut', 'manager_type', 'time_type', '서비스시작_하루', '서비스종료_하루' ,'고객가입_year', '고객가입_month', '고객가입_day', '고객가입_요일', ,'customer_type','kmeans_manager','kmeans_customer','region_type')

3. Encoding & Scaling

방식	적용 columns	기타
LabelEncoder	'결재형태', '지역'	
Mean Encoding	'장기서비스여부', '기존고객여부', '결재형태', '주거형태', '평수', '부재중여부', '우선청소', '쿠폰사용여부', '매니저사용휴대폰', '매니저이동방법','CS교육이수여부', '청소교육이수여부', '부재중서비스가능여부', '추천인여부', '서비스요일','서비스월', '매니저연령대', '부재중_일치', '반려동물여부', '접수시간', '서비스연도', '서비스계절','교육여부','이용연도', '지역','서비스분기','house_type', '매니저연령_qcut', '서비스이용기간_qcut', '서비스진행비율_qcut', '회차별일자_qcut', 'manager_type', 'time_type', '서비스시작_하루', '서비스종료_하루' ,'고객가입_year', '고객가입_month', '고객가입_day', '고객가입_요일', ,'customer_type','kmeans_manager','kmeans_customer','region_type'	- Encoding한 값을 새로운 열로 만들어줌 - test데이터의 결측값은 train 데이터의 0으로 채워줌
One hot encoding	'장기서비스여부', '기존고객여부', '주거형태', '평수', '부재중여부', '우선청소', '쿠폰사용여부', '매니저사용휴대폰', '매니저이동방법', 'CS교육이수여부', '청소교육이수여부', '부재중서비스가능여부', '추천인여부', '서비스요일','지역_매칭', '서비스월', '매니저연령대', '부재중_일치', '반려동물여부', '서비스진행비율', '서비스연도', '서비스계절', '이용연도', '서비스시작시각', '교육여부','서비스분기','서비스DAY','house_type','manager_type','time_type' ,'서비스시작_하루','서비스종료_하루','customer_type','region_type'count', '매니저_교육이수정도별count',	- Test one hot encoding에 밤_시작, 새벽_종료 열을 추가 - PCA 진행

	'이용연도별count', '회차별일자_qcut별count'	
StandardScaler	'전체회차', '현재회차', '서비스시작시간', '서비스종료시간', '고객가입_year', '고객가입_month', '고객가입_day', '고객가입_요일', '총서비스시간', '서비스-접수-차이', '매니저연령', '매니저연령_qcut', '서비스이용기간', '회차별일자', '가입_최초서비스', '접수시간', '서비스진행비율_qcut', '서비스_접수_차이_qcut', '서비스이용기간_qcut', '접수시간_qcut', '가입_최초서비스_qcut', '회차별일자_qcut', '가입_접수_차이', '접수주기', '매니저가입이후년차', '장기서비스여부별 count', '기존고객여부별 count', '결재형태별 count', '주거형태별 count', '평수별 count', '부재중여부별 count', '우선청소별 count', '쿠폰사용여부별 count', '매니저사용휴대폰별 count', '매니저이동방법별 count', 'CS 교육이수여부별 count', '청소교육이수여부별 count', '부재중서비스가능여부별 count', '추천인여부별 count', '서비스요일별 count', '서비스월별 count', '매니저연령대별 count', '부재중_일치별 count', '반려동물여부별 count', '접수시간별 count', '서비스연도별 count', '서비스계절별 count', '교육여부별 count', '이용연도별 count', '지역별 count', '서비스분기별 count', 'house_type 별 count', '매니저연령_qcut 별 count', '서비스이용기간_qcut 별 count', '서비스진행비율_qcut 별 count', '회차별일자_qcut 별 count', 'manager_type 별 count', 'time_type 별 count', '서비스시작_하루별 count', '서비스종료_하루별 count', '고객가입_year 별 count', '고객가입_month 별 count', '고객가입_day 별 count', '고객가입_요일별 count', 'customer_type 별 count', 'kmeans_manager 별 count', 'kmeans_customer 별 count', 'region_type 별 count'	-

4. 불필요한 column 제거

'접수시각','접수일','최초서비스일','서비스일자','서비스주소','고객가입일','매니저생년월일','매니저성별','근무가능지역','반려동물','매니저주소', '매니저최초가입일', '매니저최초서비스일'

5. Feature Selection

XGBClassifier를 활용하여 SelectPercentile로 Feature Selection을 진행함

5. ADASYN

ADASYN를 사용하여 오버샘플링을 진행함

Ⅱ. submission_3

1. 결측치 제거

Columns	fill value	기타
반려동물	"정보없음"	
우선청소		
매니저사용휴대폰		
매니저이동방법		
부재중여부	0	
CS교육이수여부		
청소교육이수여부		
결재형태	신용카드	train 결재형태의 최빈값
평수	2	{ '10평대' : 0, '20평대' : 1, '30평대' : 2, '40평대이상' : 3 } 로 mapping 후 2로 채움
고객가입일	test['최초서비스일'] - (train['서비스일자'] - train['고객가입일']).mean()	

2. Feature generation

1) ~ 35) submission_1의 feature와 같음

3. 불필요한 column 제거

'접수시각','접수일','최초서비스일','서비스일자','서비스주소','고객가입일','매니저생년월일','매니저성별','근무가능지역','반려동물' ,'house_type'

4. Encoding & Generation

방식	적용 columns	기타
LabelEncoder	'결재형태','주거형태','우선청소','매니저사용휴대폰','매니저이동방법','서비스계절','부재중_일치', 'house_type','manager_type' ,'cus_type'	
Interaction Features	Column들 중 고유값이 25개 이하인 35개의 Feature들을 Categorical Feature로 판단 Categorical Feature들에 자기 자신과 다른 Categorical	

	Feature하나의 조합으로 Mean Target Encoding을 실행	
Polynomial Features	Column들 중 고유값이 25개 초과인 44개의 Feature들을 Numerical Feature로 판단. Numerical Feature들에 대해 PolynomialFeatures를 사용해 수치형 변수 사이에도 교호작용 변수 추가	

5. Remove Perfect Collinearity

Pycaert 내장의 setup environment를 사용해 Perfect Collinearity 제거

6. ADASYN

ADASYN를 사용하여 오버샘플링을 진행함