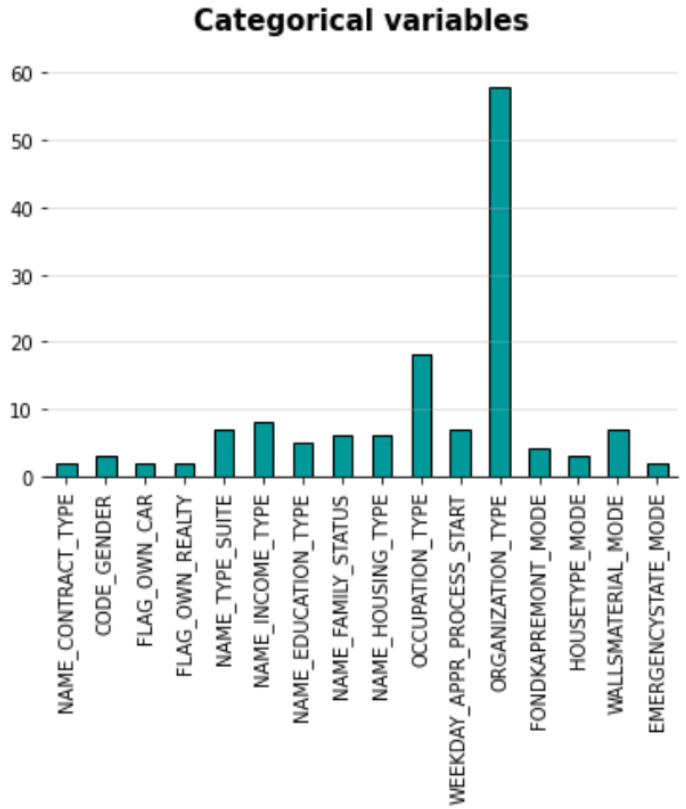
Week 8 스터디 발표

필사 Kernel과 다른 점

1. 범주형 변수의 encoding 방법

- 범주 2개(binary): One-hot encoding
- 범주 3개 ~ threshold 미만: Label encoding
- 범주 threshold개 이상: Target encoding with smoothing



2. K-fold 대신 Stratified K-fold 이용

→ TARGET 클래스의 분포가 매우 불균형함(Highly imbalanced)

→ K-fold를 사용할 경우 상대적으로 비율이 높은 0 클래스가 특정 폴드의 데이터를 전부 차지할 수 있음

→ Stratified K-fold 이용

필사 Kernel과 다른 점

3. Correlated Features

필사 Kernel에서는 highly correlated feature들을 제거했다.

→ Tree 계열 모델은 feature간 correlation이 크더라도 regression 처럼 영향을 크게 받지 않기 때문에 제거하지 않아도 무방

→ 하지만, redundant feature를 제거하는 것은 모델 성능 향상에 도움이 되기 때문에 이러한 feature들은 제거하는 것이 좋음

Overfitting

	fold	train	valid
0	0	0.857624	0.762248
1	1	0.834657	0.767332
2	2	0.818071	0.767346
3	3	0.818274	0.766577
4	4	0.821497	0.764951
5	overall	0.830025	0.765521

→ Train AUC와 valid AUC의 차이가 약 0.05 이상으로 overfitting이 발생

[해결 방안]

1. Feature 수 감소
: Overfitting 발생 이유가 curse of dimensionality 때문일 수 있으므로 적절한 Feature selection 방법을 선택하여 불필요한 feature는 제거한다.

2. Hyper parameter Tuning
: Grid Search, Random Search, Bayesian optimization 등 tuning 방법을 이용해 적절한 parameter를 찾는다.