방학 세미나

3팀

김서윤
이수경
이지연
진효주

INDEX

1

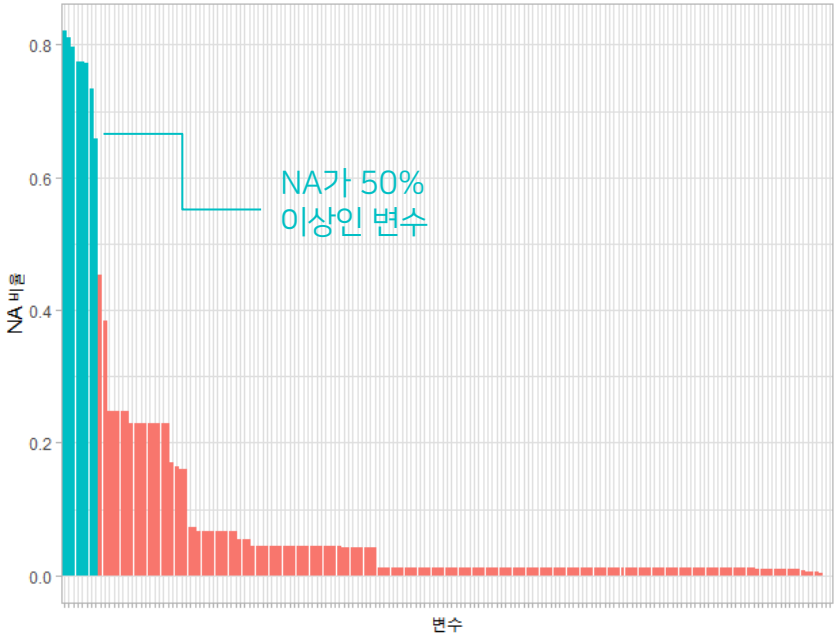
들어가며…

들어가며

데이터 탐색

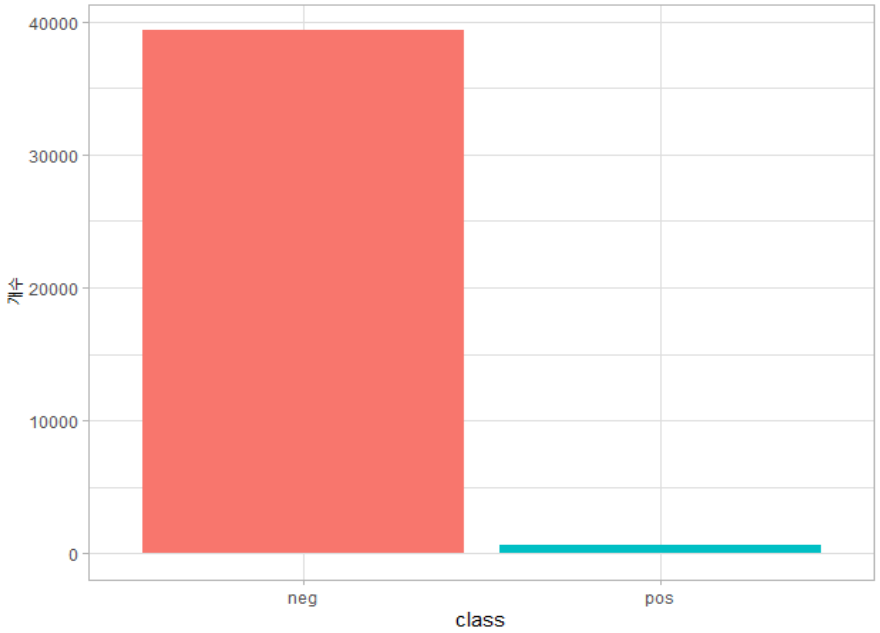
트럭의 air pressure system 에 대한 익명의 171개 변수들에 대한 정보

1. NA 가 매우 많은 데이터



변수별 NA 비율

2. 클래스 불균형 데이터



Train 데이터의 클래스 비율

들어가며

분석 흐름 (a.k.a. 수제 그리드 서치)

한땀한땀… 우리 손으로 그리드 서치 했습니다.

NA 처리

- 변수·관측치 제거
- KNN imputation
- MICE imputation

차원축소 / 클래스 불균형 처리

- 차원축소 : PCA
- 오버 샘플링 : SMOTE

모델링

- 로지스틱 회귀
- KNN classifier
- ~~SVM~~ *끝나지 않는 튜닝을 기다리다 포기…*
- Random Forest
- Light GBM

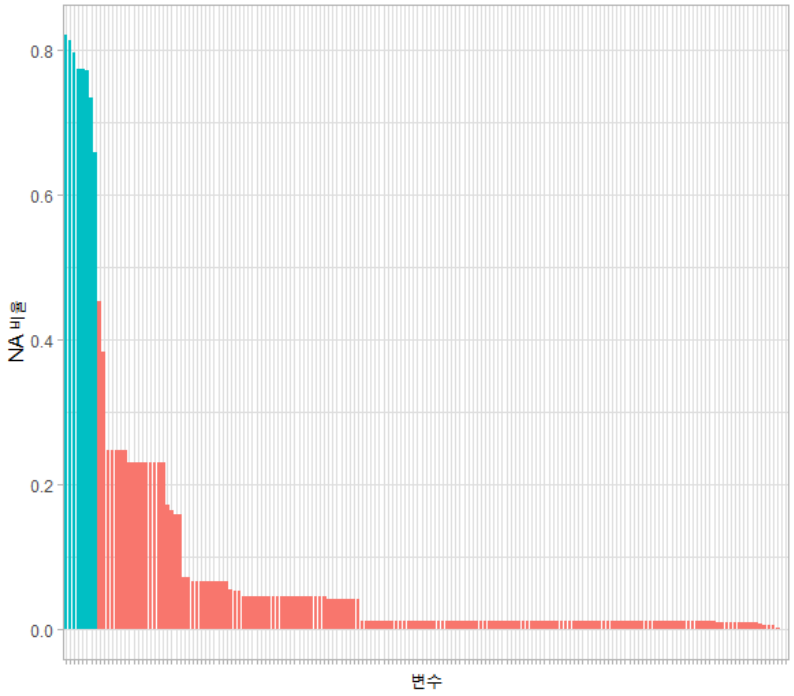
2

데이터 전처리

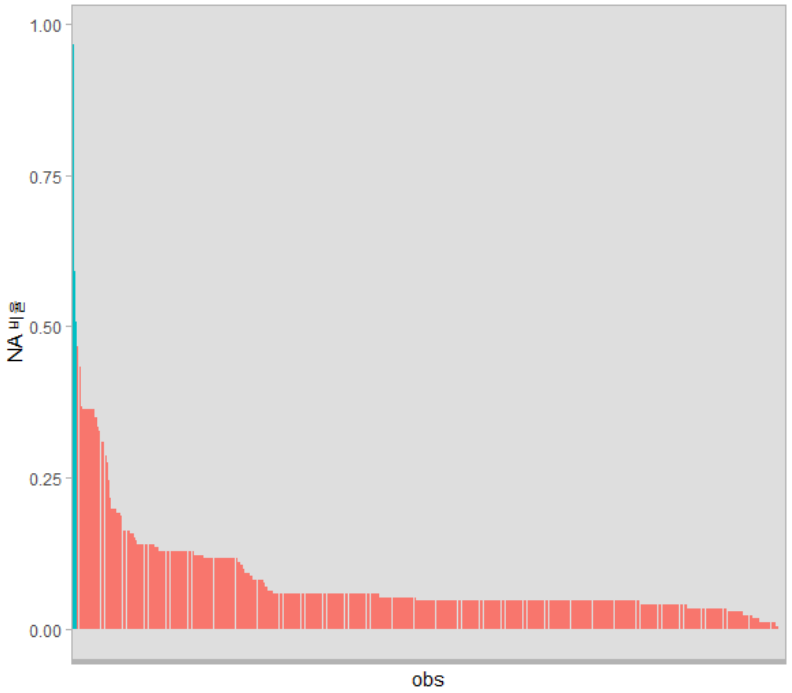
데이터 전처리

NA 처리 : 결측 비율 50% 변수·관측치 제거

결측치가 50% 이상인 변수와 관측치 제거 -> 39733 obs * 163 var



변수별 NA 비율



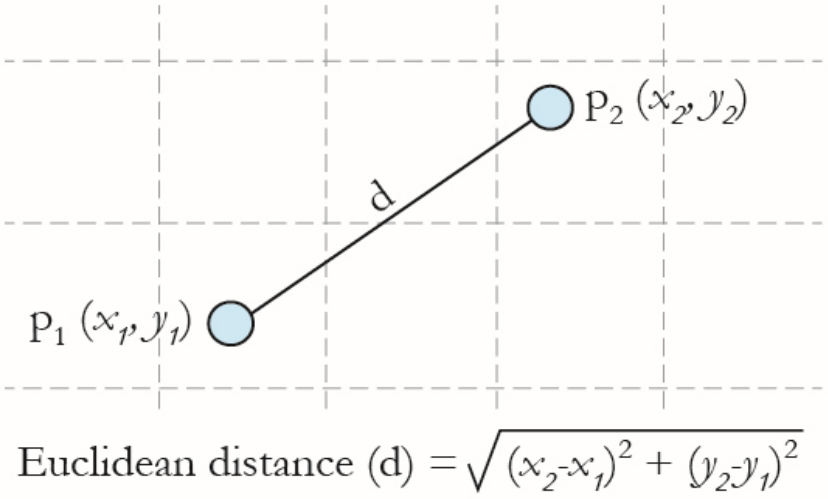
관측치별 NA 비율

데이터 전처리

NA 처리 : KNN imputation

머신러닝에서 가장 흔히 사용되는 결측치 대체 방법

KNN (K-Near Neighbors)



결측이 발생한 개체와 **가장 가까운 거리에 있는 k개의 이웃** 개체들을 활용하여 결측값을 대치하는 비모수적 방법

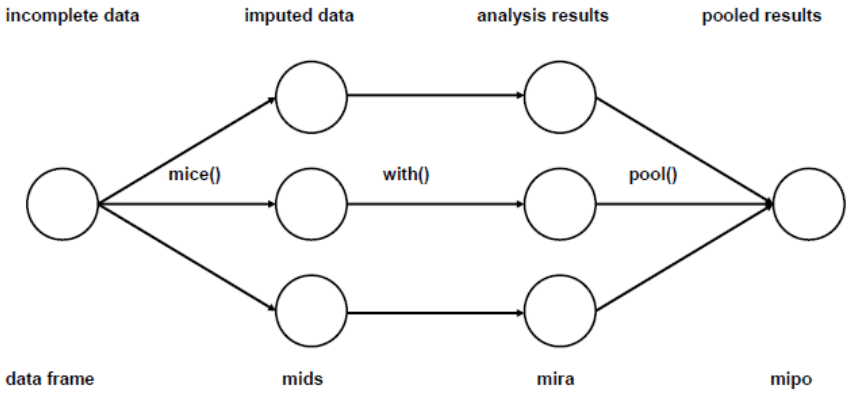
패키지 문제에서도 자주 등장했던 knn imputation… 내장된 패키지인 사이킷런을 통해 손쉽게 구현 가능하다! 다만 데이터셋이 커서 오래 걸림…

데이터 전처리

NA 처리 : MICE imputation

다중대체법(Multiple Imputation) 중 가장 많이 사용되는 방법

MICE (Multiple Imputation by Chained Equations)



변수가 여러 개인 데이터에 대하여 한 변수 내 결측치를 **다른 변수들로 모형화**하고 이를 **연쇄적으로 교대**하여 결측치를 대체하는 방식

5개 데이터셋 maxit=5 반복했다가 15시간 걸렸던… 레전드 코드…

근데 pooling 안되고 꺼져서 complete 1만 사용함…

2 데이터 전처리

차원축소 : PCA

MICE Impute Data

95%의 설명력을 지니는 77개의 변수로 차원 축소

	PC74	PC75	PC76	PC77
Standard deviation	0.58356	0.57591	0.57047	0.56896
Proportion of Variance	0.00224	0.00218	0.00214	0.00213
Cumulative Proportion	0.94266	0.94484	0.94698	0.94911
	PC78	PC79	PC80	PC81
Standard deviation	0.55969	0.55717	0.55224	0.54527
Proportion of Variance	0.00206	0.00204	0.00201	0.00196
Cumulative Proportion	0.95117	0.95321	0.95522	0.95718

KNN Impute Data

95%의 설명력을 지니는 9개의 변수로 차원 축소

	0	1	2	3	4	5	6	7	8
0	1.000000	-0.000000	0.000000	-0.000000	-0.000000	-0.000000	-0.000000	0.000000	0.000000
1	-0.000000	1.000000	0.000000	0.000000	-0.000000	0.000000	-0.000000	0.000000	-0.000000
2	0.000000	0.000000	1.000000	0.000000	-0.000000	0.000000	-0.000000	-0.000000	-0.000000
3	-0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	-0.000000	-0.000000
4	-0.000000	-0.000000	-0.000000	0.000000	1.000000	0.000000	0.000000	-0.000000	0.000000
5	-0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	-0.000000
6	-0.000000	-0.000000	-0.000000	0.000000	0.000000	0.000000	1.000000	-0.000000	0.000000
7	0.000000	0.000000	-0.000000	-0.000000	-0.000000	0.000000	-0.000000	1.000000	-0.000000
8	0.000000	-0.000000	-0.000000	-0.000000	0.000000	-0.000000	0.000000	-0.000000	1.000000

클래스 불균형 처리 : SMOTE

1. 소수 클래스의 데이터 중 하나를 선택하여 가장 가까운 소수 클래스 데이터를 k개 선택
2. 처음 선택한 데이터와 k개의 데이터 사이의 직선을 그리고 직선 상에 가상의 소수 클래스 데이터 생성

데이터 전처리

클래스 불균형 처리 : SMOTE

MICE Impute Data

1 MICE + PCA X

2 MICE + PCA O

KNN Impute Data

3 KNN + PCA X

4 KNN + PCA O

4개 데이터 모두에
SMOTE로 오버 샘플링

3

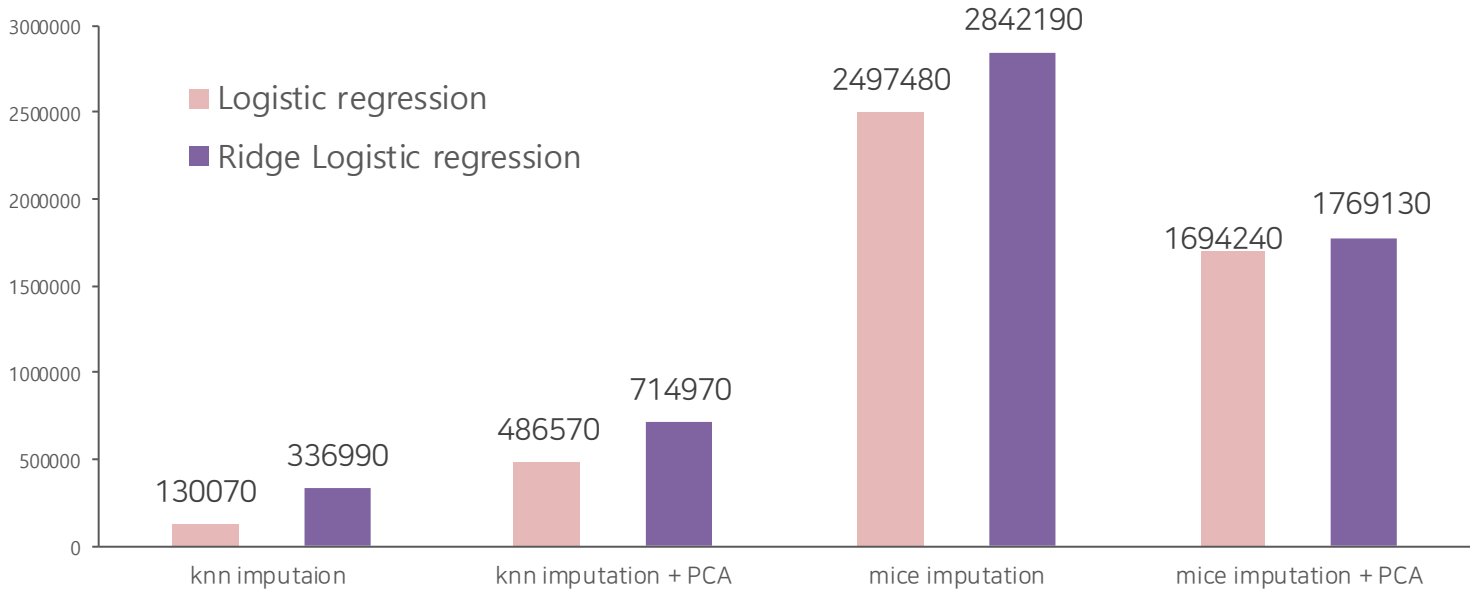
모델링

모델링

Logistic Regression & Ridge Logistic Regression

- 이항 분류에서 가장 대표적인 모델인 로지스틱 회귀 사용
- 변수의 개수가 많으므로 overfitting 방지를 위해 ridge penalty 준 로지스틱 회귀 사용

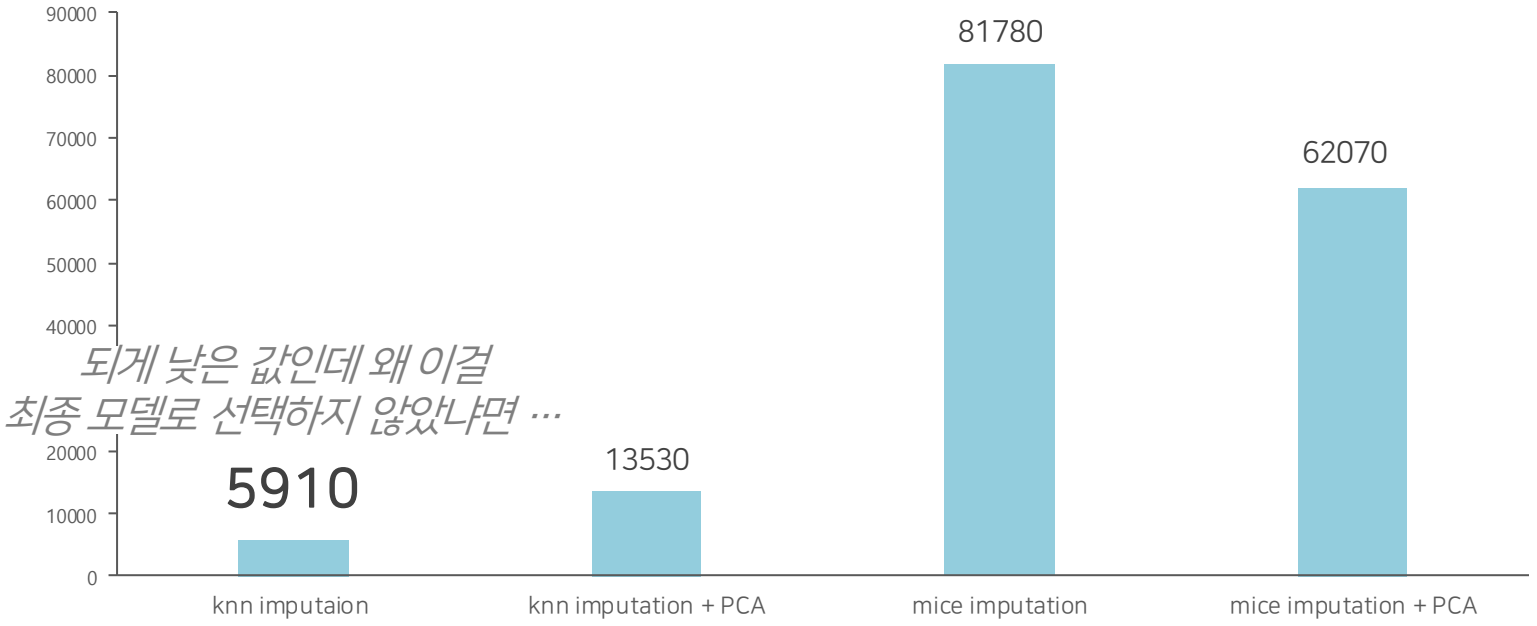
** 모델의 성능은 hold-out 검증을 통해 30% test set 활용하여 계산한 cost로 비교*



모델링

K-Nearest Neighbor

- Imputation에서 썼던 것과 동일한 알고리즘
- But, 이번에는 학습한 데이터의 class값을 바탕으로 예측 값을 분류



모델링

K-Nearest Neighbor → 왜 안 썼는가..?!

< Cost를 최소화 하는 최적의 K 값 >

KNN + PCA X

	0
0	-2070.0
2	-3160.0
1	-4450.0
4	-5324.0
3	-6126.0
6	-7458.0
5	-8106.0
8	-8568.0
7	-9358.0
9	-10962.0

MICE + PCA X

	0
0	-21202.0
2	-23544.0
4	-29124.0
3	-32936.0
6	-33494.0
8	-38224.0
5	-38284.0

KNN + PCA O

	0
0	-2248.0
2	-4586.0
4	-6060.0
1	-6366.0
3	-6876.0
...	...
228	-93572.0
227	-93664.0
230	-93874.0
229	-94266.0
231	-94766.0

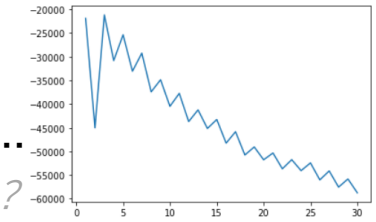
MICE + PCA O

	0
2	-21148.0
0	-21898.0
4	-25350.0
6	-29240.0
3	-30818.0
5	-33064.0
8	-34868.0
7	-37418.0
10	-37740.0

KNN algorithm은
고차원 데이터에는 부적합

최적의 K가 자꾸
0 또는 2래서 신뢰가 안 갔음..

왜 이러는지 아시는 분 알려주세요..?

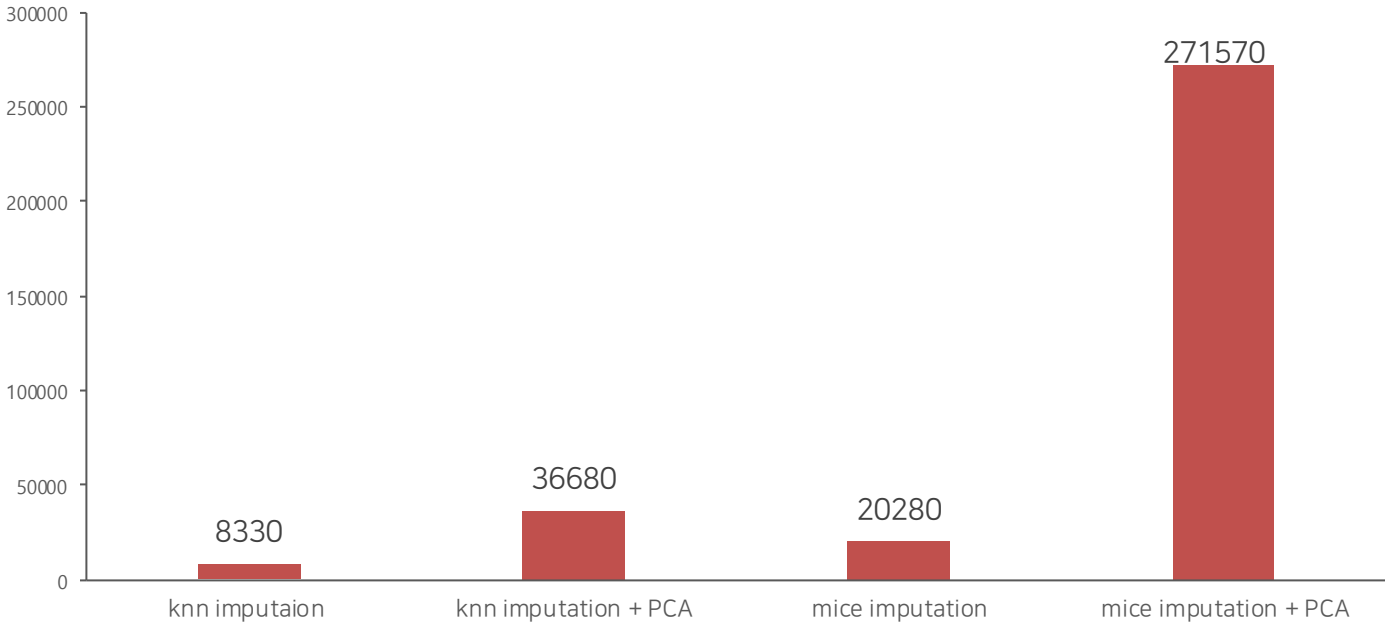


모델링

Random Forest

- 앙상블 모델의 대표적인 기법인 Random Forest 이용
- KNN imputation에서의 cost가 8330으로 제일 적음을 확인

** tree 개수는 50개로 지정, 시간 단축을 위해서 ranger 이용*

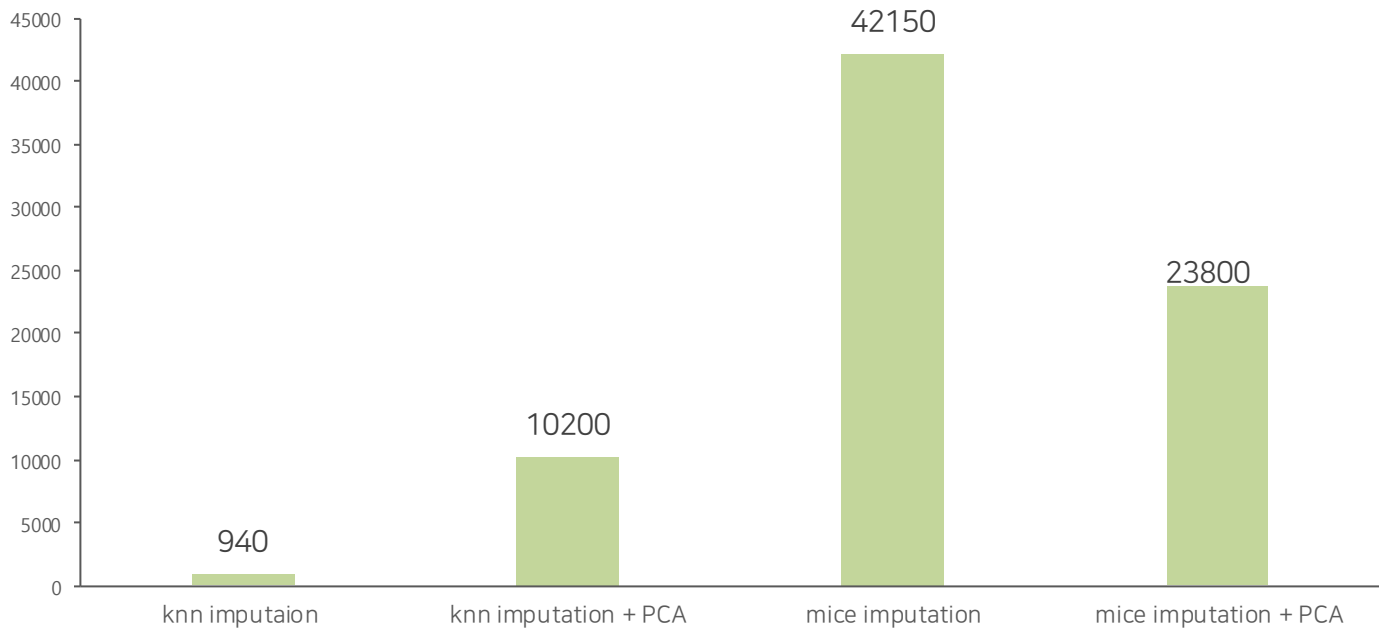


** PCA 안 했을 때의 cost가 더 적다 !!*

모델링

Light GBM

- 트리 기반 학습 알고리즘인 Gradient boosting 모델로 속도가 빠른 장점이 있음
- 마찬가지로 KNN imputation 에서 940으로 가장 적은 것을 확인(!!)



4

마치며…

4 마치며

최종 모델 선정…?

최종 데이터셋	최종 모델
knn imputation	**Random Forest**

30% test set으로 계산한 최소 cost

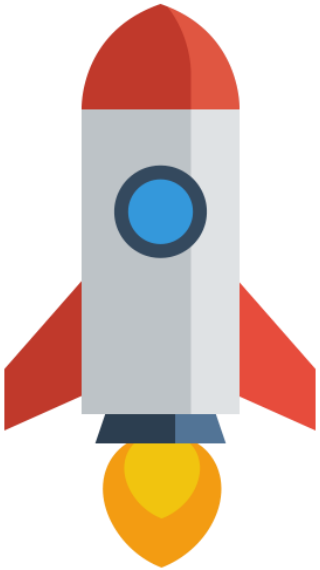
$8330

3팀의 효자… 나무자람…

인줄 알았는데…

4 마치며

최종 모델 선정



최종 데이터셋	최종 모델
knn imputation	**Light GBM**

LightGBM

" 30% test set으로 계산한 최소 cost

$940 "

마지막날까지 팀안에서 캐글 컴피티션하기..

나무자람을 기특하게 여기고 ppt까지 만들었으나 LGBM이 더 좋은 성능을 내었다…!

4 마치며

한계와 의의

- ✓ 기본 10시간이 넘는 NA대체와 모델 파라미터 튜닝 코드
- ✓ 나무자람에는 인내심이 필요하다
- ✓ 온라인 세미나로 직접 만나지 못한 아쉬움

- ✓ 여러 전처리 방식과 모델을 사용하여 최소의 비용 찾는 손(hand)튜닝 진행
- ✓ NA값 대체, 차원축소, 클래스 불균형을 위한 샘플링, 등 모든 과정이 지난 클린업·주제분석에서 배운 내용들을 복습하고 직접 사용해볼 수 있었던 좋은 기회!
- ✓ 팀원 모두 한 개 이상의 모델을 맡아 모델링
- ✓ R과 Python 모두 사용하여 언어의 장벽을 뛰어넘는 감동 실화

4 마치며

왠지..한마디…

일주일이라는 짧은 시간동안이었지만
귀여운 친구들과 함께해서 햄보캣다 짱 !! >.<
나무가 자라는게 정말 재미있었고..^^
노트북 열 식혀주느라 계속 손풍기를 노트북한테
쏴드렸는데… 마치 노트북을 모시는
하인이 된 것 같은 기분이었다.. 굿..~~

만나서 하지 못해 아쉬웠지만 랜선으로
머신러닝과 친목도모를 둘다 해낸 갓팀…
1등 못해도 곱창을 먹을 것~~ 예측 성능
높이기만을 위한 모델링은 처음이었는데
알고있다고 생각했는데 사실은 거짓 알고있음
이었던 것들을 다시 공부하면서 부족한 부분을
많이 알아갔던..뜻깊은 일주일이었따..^__^

웹엑스 화면으로 보던 귀요미 친구들과 함께한
일주일이라 너무 뿌듯했다,, !!!
오프라인 만남을 하지 못해서 아쉬울 뿐이고
ㅠㅠ… 다들 너무너무 고생했고
학기 시작해서 다른 팀이 되더라도 ,, ㅠㅠ
언제나 행복하렴 귀여미들아 !! (뒤풀이때 보쟈)
마지막으로 방세 마무리의 영광을,,,
일주일동안 잠든 적 없는
나의 그램 2020 14인치에게 돌립니다…

일주일동안 착하고 귀여운 똑똑이들과
같은 팀을 해서 좋아따>< 초반에 바빠
서 하루종일 참여하진 못했는데 이해해
준 친구들 덕분에 너무 고마웠구 더 열
심히 참여할 수 있었던 거 같다 흑흑 ㅠ..
비록 온라인이었지만 다들 열심히 했구
우리팀 최고다 얘두라 고마워 곱창 먹쟈
~~~!
~~~

THANK YOU