



수면관리를 통한 비만관리

1조

A photograph of two women, one of Asian descent and one of African descent, standing against a solid yellow background. They are both wearing grey long-sleeved shirts. The woman on the right is holding a large, bright yellow rectangular sign. The sign has text written on it in two different styles: orange block letters and black cursive script. The text on the sign reads '해결하고자 하는 Business Problem? #Obesity'.

해결하고자 하는
Business Problem?

#Obesity

해결하고자하는 Business Problem

비만인 사람들을 위한 적절한 가이드라인을
제공함으로써
사용자의 삶의 변화를 이끌어 냄

비만인 사람들의 삶에서 발견할 수 있는
공통적인 요소를 발견하여 패턴을 찾음

‘손목닥터 9988’을 사용하는 비만 사용자들에게도
더 나은 라이프 스타일, 올바른 삶의 패턴을 제안
하여 비만 문제를 해결




사용할 Dataset :

Sleep Health and Lifestyle Dataset

비만에 영향을 미치는 요인들을 분석할 수
있는 적절한 데이터가 필요함

‘Sleep Health and Lifestyle Dataset’은 한 사람의
BMI 카테고리, 수면 시간,
수면의 질, 스트레스 지수 등
비만 요인의 후보가 될 수 있는
많은 데이터를 갖고 있음

이러한 데이터의 속성들을 활용하여 단순히 비만과의
상관관계만을 분석하는 것이 아니라
다른 속성들과의 상관관계도 분석함으로써
더 정확한 패턴을 찾을 수 있음

Gender	Age	Occupation
String	Integer	String
Male 51%		Nurse 20%
Female 49%		Doctor 19%
		Other (230) 61%
Male	27	Software Engineer
Male	28	Doctor
Male	28	Doctor
Male	28	Sales Representative
Male	28	Sales Representative
Male	28	Software Engineer
Male	29	Teacher
Male	29	Doctor

Modeling

01

Feature Selection

Filter approach

ElasticNet

Forward, Backward selection

02

Hyperparameter tuning

Cross validation

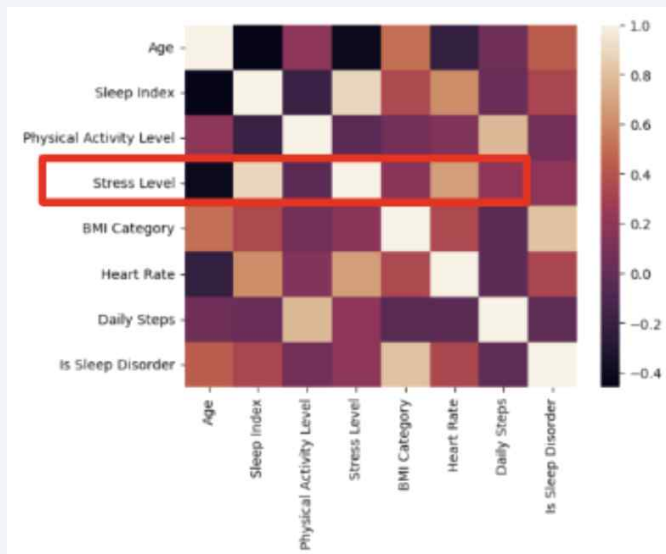
Random Forest

03

Identify most profitable segments

Modeling

Feature selection-Filter approach(1)



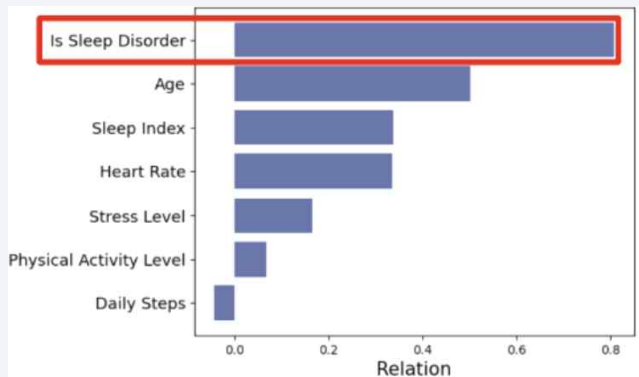
히트맵을 통해 Target인 BMI Category와
다른 attribute간의 상관계수를 계산
Feature selection의 여러 방법 중
모델링 전에 골라내는 Filter Approach를 먼저 사용



Target과 연관성이 높은 attribute만 사용함으로써
모델 학습 속도를 높이고,
over-fitting도 방지할 수 있음

Modeling

Feature selection-Filter approach(2)

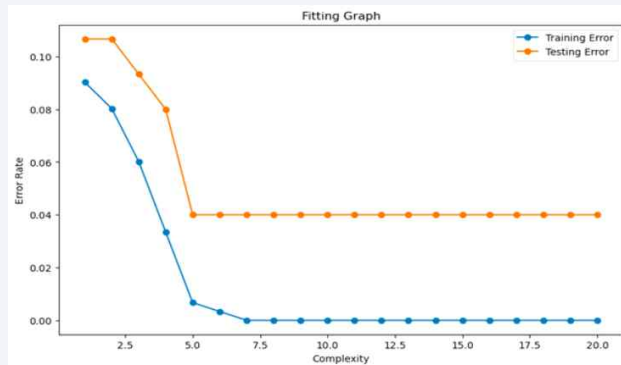
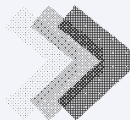
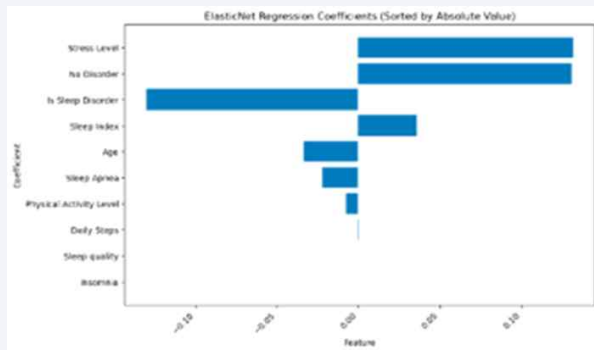


상관계수를 계산해본 결과,
'Is sleep disorder'(수면장애 여부)가 약 0.8로 target인 BMI category와 가장 강한 linear relationship을 보임
반면 'physical activity level', 'daily steps'는 상대적으로 약한 linear relationship을 보임

하지만 decision tree는 non-linear한 관계도 예측가능하기 때문에
최종 feature attribute들을 고르기 위해서는 추가적인 분석이나
직접 여러 모델들을 시험해 보는 것이 필요

Modeling

Feature selection-ElasticNet

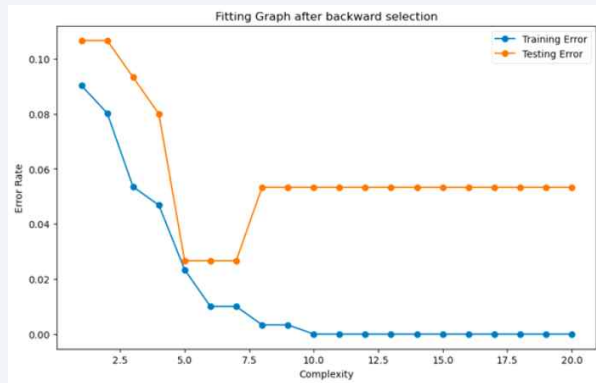
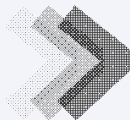
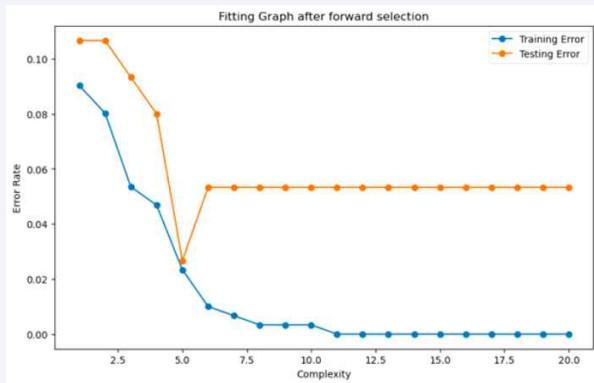


sample 수가 적을 때 사용할 수 있는
elasticNet를 사용해서 feature selection을 진행하였다.
GridSearchCV를 통해 최적의 max_iter와 alpha 값을 찾아
Regression을 진행하고 coefficient의 절댓값을
기준으로 정렬하였다.

상위 5개의 값을 선택하여
training을 진행하였고,
결과적으로 sweet spot에서의 accuracy가
향상되었다.

Modeling

Feature Selection-Forward, Backward



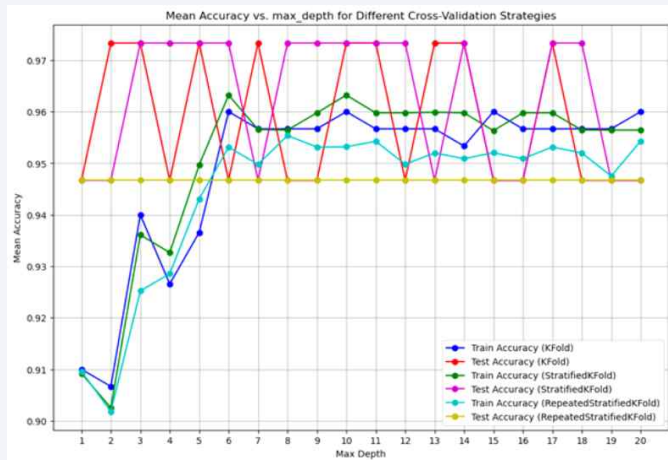
wrapper method인

forward, backward selection을 진행하여
elasticNet으로 feature selection 하였을 때와 비교

결과적으로 backward selection을 통해
선택된 feature들로 training을 진행하였을 때
**Complexity와 상관없이 전반적으로
accuracy 값이 향상되었고,
sweet spot에서의 성능도 더 좋아졌음**

Modeling

Hyperparameter tuning-Cross Validation



Cross Validation: model validation의 종류 중 하나

모델 검증 또는 overfitting 방지를 위해 dataset을 분할하여 학습할 때 사용하는 방법이지만, 하나의 dataset을 랜덤한 여러개의 dataset으로 분할하여 학습을 진행함으로써

성능이 뛰어난 **hyperparameter**를 찾을 수 있다는 점에서 **hyperparameter tuning**에 활용할 수 있었음

대표적인 k-fold, stratified k-fold, repeated stratified k-fold를 사용해서 진행하였고, 최적의 max_depth 값을 찾을 수 있었지만, 결과적으로 성능이 좋아지지 못하여 활용할 수 없었음

Modeling

Hyperparameter tuning-RandomForest

```
검증 세트 정확도: 0.97  
하이퍼파라미터 튜닝 후 최적 모델: RandomForestClassifier(max_depth=10, min_samples_split=6, n_estimators=130,  
                                                             random_state=42)  
최적 파라미터: {'n_estimators': 130, 'min_samples_split': 6, 'min_samples_leaf': 1, 'max_depth': 10}
```

이번에는 가장 대표적인 방법들로

Hyperparameter tuning을 사용해보기로 하여
Tuning과 동시에 **모델 자체의 성능** 또한 향상시키기 위해
RandomizedSearchCV와 함께
RandomForest 방식으로 tuning과 학습을 함께 진행했음

hyperparameter tuning 시에는
overfitting 방지를 위해
max_samples_split과
min_samples_leaf도 제한하였고,
결과적으로 도출된
hyperparameter 들로 training
진행 결과 trainingset과 testset에
대한 accuracy 모두 증가했다.

Modeling

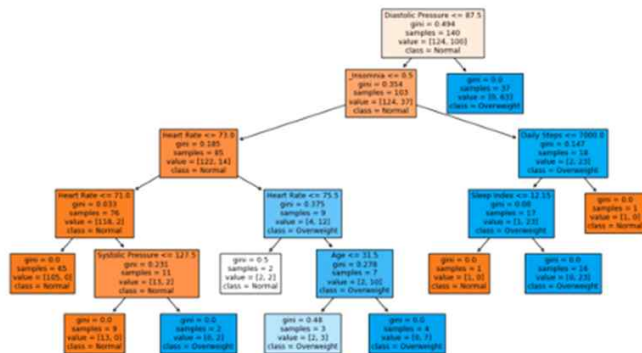
Hyperparameter tuning-RandomForest

검증 세트 정확도: 0.97

하이퍼파라미터 튜닝 후 최적 모델: RandomForestClassifier(max_depth=10, min_samples_split=6, n_estimators=130,

random_state=42)

최적 파라미터: {'n_estimators': 130, 'min_samples_split': 6, 'min_samples_leaf': 1, 'max_depth': 10}



모델링의 feature attribute으로 사용된
불면증 여부는 추후에 손목닥터 9988에서
측정되는 수면 중 깨어 있는 시간,
얕은 수면 시간, 깊은 수면 시간의
3가지 지표를 활용하여
적절하게 반영해야 할 것임

Modeling

Identify most profitable segments

entropy = 0.696
samples = 16
value = [3, 13]
class = Overweight

Segment 1

entropy = 0.503
samples = 36
value = [4, 32]
class = Overweight

Segment 2

entropy = 0.0
samples = 96
value = [0, 96]
class = Overweight

Segment 3

	Normal instance 수	Overweight instance 수	Overweight 비율
Segment 1	3	13	81.25%
Segment 2	4	32	88.88%
Segment 3	0	96	100%

프로젝트에서 목표로 해야 할 segment:
class가 'overweight'로 분류되는 segments

즉, 수면습관 개선, 적정 수준의 운동 등을 통해
과체중인 사람을 정상체중이 될 수 있게
유도할 수 있음

이 중에서도 목적에 더 부합하는 segment:
overweight 비율이 더 높은 segment

Modeling

Identify most profitable segments

entropy = 0.696
samples = 16
value = [3, 13]
class = Overweight

Segment 1

entropy = 0.503
samples = 36
value = [4, 32]
class = Overweight

Segment 2

entropy = 0.0
samples = 96
value = [0, 96]
class = Overweight

Segment 3

프로젝트 비용을 효율적으로 쓰기 위해서는
overweight 비율이 가장 높은 Segment 3에
집중하는 것이 효율적일 수 있음

	Normal instance 수	Overweight instance 수	Overweight 비율
Segment 1	3	13	81.25%
Segment 2	4	32	88.88%
Segment 3	0	96	100%

예시에서는 비율 차이가 크게 차이 나지 않지만,
비율 차이가 더 나는 경우에는 투자 비용에 대한
가치를 극대화할 수 있음

Evaluation

01

Fitting Graph

Depth를 증가시키며
Accuracy 비교

02

ROC-AUC Curve

임계값을 변화시키며 분류
문제에 대한 성능 측정

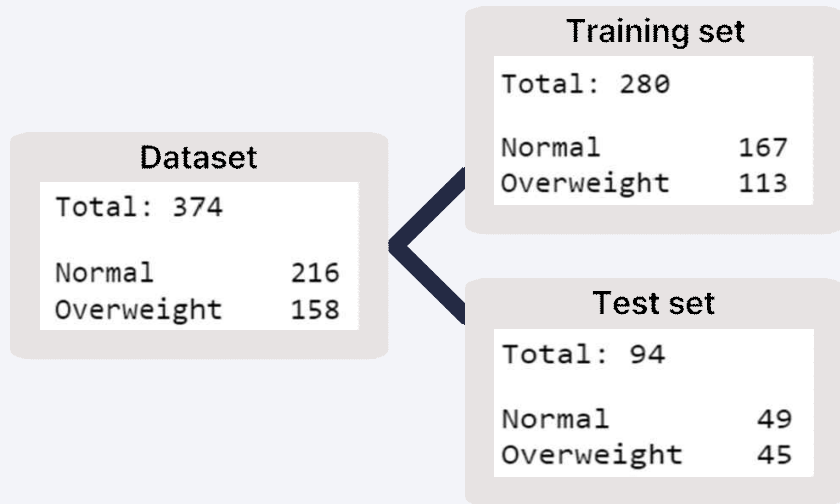
03

Confusion Matrix

분류모델의 성능 지표를 계산

Evaluation

Training set/Test set split



모델의 일반화 성능을 확인하기 위해
전체 Dataset의 75%: Training set
나머지 25%: Test set

Training set: 모델 생성 시
Test set: Confusion matrix 및 모델 평가
에 사용됨

Evaluation

Training set/Test set split

Test set

Total: 94

Normal 49

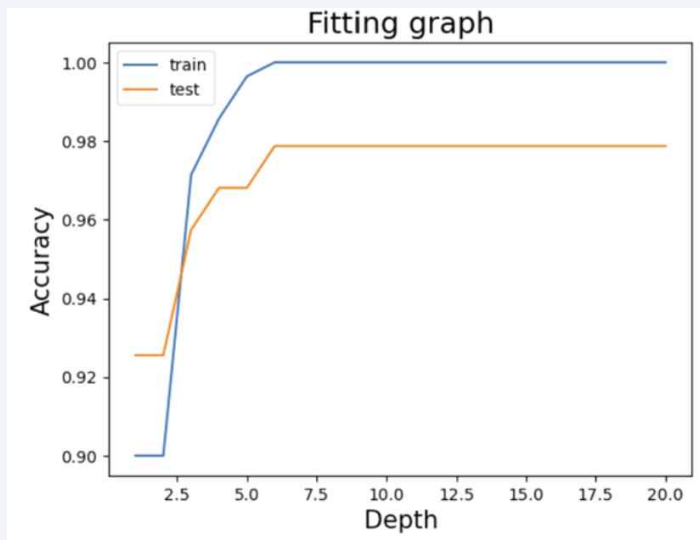
Overweight 45

Test set의 distribution을 살펴보면,
Normal label이 49개,
Overweight인 label이 45개로
비슷하게 분포되어 있음

즉, 도출된 모델로 BMI category를
normal로 구별할 때와 overweight으로 구별할 때
양쪽 모두의 성능을 충분히 살펴볼 수 있을 것이다.

Evaluation

Fitting Graph



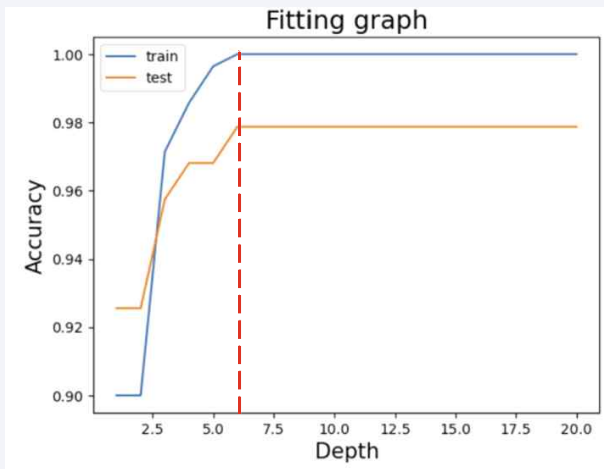
최적의 모델을 찾기 위해서는 모델의 **complexity(Depth)**를 증가시키면서 training accuracy와 test accuracy를 비교하여 **Fitting graph**를 만들 수 있음

초기에는 Depth가 증가함에 따라 train accuracy와 test accuracy가 모두 증가

Evaluation

Fitting Graph

```
best_dt = tree.DecisionTreeClassifier(criterion = "entropy", max_depth = 5)
```

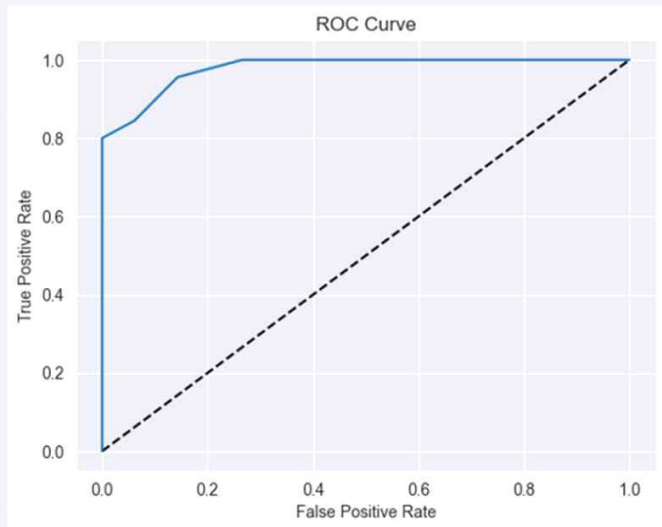


그러나 Depth가 6 이상일때는
train accuracy는 100%가 되어
overfitting이 의심되고,
test accuracy 또한 유의미하게 증가하지 않음

따라서 모델의 최대 Depth를 5로 제한하여
최적 모델을 만들

Evaluation

ROC-AUC Curve

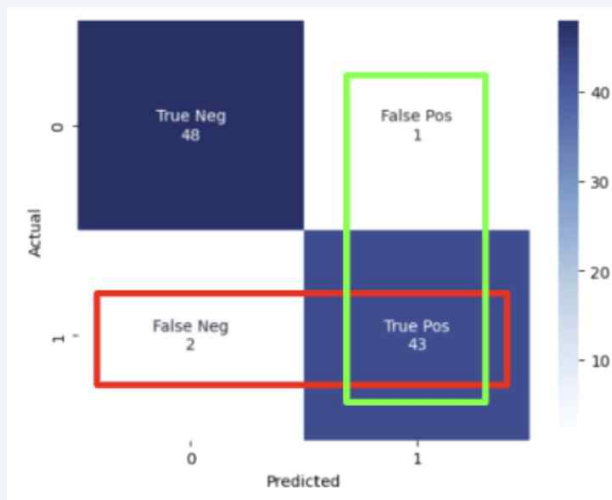


ROC-AUC Curve를 그려 평가한 결과,
AUC 점수가 0.978으로 높은 점수를 나타냄
해당 그래프로 미루어 보아, 이 모델은 hold-out
data set에서는 'normal' 과 'overweight' 사이의
상대적인 차이를 잘 포착할 수 있음
그러나 모델을 실제 현장에 적용했을 때 원하는 만큼
정확도가 나오지 않을 수 있음

따라서 실제 현장에서 추가적으로 얻게 되는 데이터를
바탕으로 분석에 사용되는 feature attribute을
변경하거나 모델의 세부적인 부분을 추가적으로
조정하는 것이 필수적

Evaluation

Confusion Matrix



Test data에 대하여 Confusion Matrix를 만들고 이를 통해 Accuracy, Recall, Precision을 계산할 수 있음

$$\text{Accuracy} = (48+43)/(48+43+2+1) = 97\%$$

$$\text{Recall} = \text{TP}/(\text{TP}+\text{FN}) = 43/(43+2) = 96\%$$

$$\text{Precision} = \text{TP}/(\text{TP}+\text{FP}) = 43/(43+1) = 98\%$$

True Pos: 실제로 Overweight인 사람을 Overweight로 예측하는 경우

Evaluation

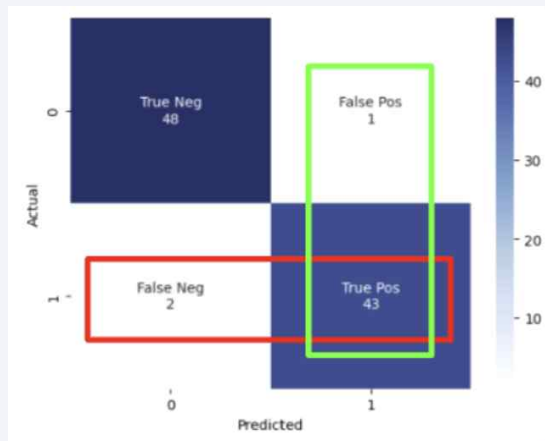
Confusion Matrix

$$F_{\beta} = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$$

이 모델은 비만인 사람들을 구별해 내고
그들에게 적절한 가이드라인을 제공함으로써
사회 전반의 건강을 증진시키는 것이 최종 목표이므로
비만인 사람들을 최대한 많이 비만으로 분류하는 것이 관건

따라서 실제로 과체중인 사람들 중 얼마나 많은 사람을
비만으로 구분할 수 있는지를 나타내는 **recall**을
precision보다 더 중요하게 볼 필요가 있음

모델의 성능 평가를 위해 F-score를 활용할 때 **recall**에
더 가중치를 두는 **F2-score**를 사용하는 것이 더 좋을 것이다.



THANK YOU

감사합니다