클린업 3주차 패키지

- 분석 툴은 R/Python 둘 다 가능합니다. 이번 패키지 문제의 조건 및 힌트는 Python을 기준으로 하지만, R을 사용해도 무방합니다.
- 제출형식은 HTML, PDF 모두 가능합니다. **.ipynb 이나 .R 등의 소스코드 파일은 불가능합니다.** 파일은 psat2009@naver.com으로 보내주세요.
- 패키지 과제 발표는 세미나 쉬는 시간 후에 하게 되며, 랜덤으로 16시 00분에 발표됩니다.
- 제출기한은 **목요일 23:59까지** 이고, 지각 시 벌금 5000원, 미제출시 10000원입니다. 패키지 무단 미제출 2회 시 퇴출이니 유의해주세요. 또한 불성실한 제출로 인한 경고를 2회 받으실 시도 퇴출이니 유의해주세요.

Chapter 1. Categorical Feature Encoding, Modeling

이번 챕터의 주요 과제는 Feature Engineering입니다. 그 중에서도 범주형 Feature에 대해 자세히 다뤄볼 예정인데요. 모델 학습을 효과적으로 진행하기 위해서는 문자 형태로 존재하는 변수를 수치형으로 바꿔주어야 합니다. 또 가령 범주형 변수가 만약 숫자로 나타나 있다고 하더라도, 그것이 순서나 수치를 의미하지는 않기 때문에 이 또한 적절한 값으로 바꿔줘야 할 것입니다. 이러한 과정을 Encoding이라고 하고, 이때 Data Leakage가 발생하지 않도록 주의해야 하기도 합니다. Data Leakage는 2주차 클린업 때 제공했던 자료와 설명을 참고해주세요.

이번 주차는 Data Leakage에 유의하며, 적절한 Feature Engineering을 수행하여 예측 성능을 높일 수 있는 방법을 배워보도록 하겠습니다.

문제1. data_week3.csv 파일을 불러와 train에, test_week3.csv를 불러와 test에 저장해주세요.
(참고 : 이번 데이터는 클린업 1주차에서 다루었던 데이터와 기본적으로 같으나, 과제의 원활한 진행을 위해 약간의 수정이 가해진 데이터입니다. 필요시 1주차 클린업 과제에서 수행했던 시각화를 참고하세요.)

문제2. train에서 변수 별 결측치의 수와, unique 값의 개수, 자료형이 나타나도록 데이터프레임을 만들어주세요. 예시는 아래와 같습니다.

	결측치 수	고유한 값 수	자료형
Comp	0	28	object
Runned	0	264	int64
Fuel Type	0	5	object
Transmission	0	2	object
Color	0	16	object
Seller Type	0	2	object
Engine	16	73	float64
Length	13	143	float64
Width	13	108	float64
Height	13	120	float64
Seat Capa	13	6	float64
Fuel Capa	24	39	float64
Age	0	18	int64

(BONUS) 데이터프레임을 parameter로 받는 함수를 선언해 문제2를 수행할 수 있도록 해주세요.

문제3. 클린업 1주차 패키지과제 및 수행했던 EDA들을 참고하여, 결측치를 적절한 값으로 대체해주세요.

(Caution) 데이터를 전처리하는 과정에서, Test Data에 전처리를 진행할 때는 Train Set과 동일한 전처리를 진행해줘야 합니다. 즉, Test set의 정보가 반영되는 Data Leakage가 발생해서는 안 됩니다.

문제4. 대표적인 인코딩 방법인 One-Hot Encoding과 Label Encoding에 대해 알아본 후, 각각의 장단점을 1~2줄로 작성해주세요.

문제5. 고유한 값의 개수가 많은 변수들이 많기 때문에, 모두 One-Hot Encoding을 적용하여 모델링을 진행하는 것은 힘들 것입니다. 모든 object형 변수들에 대해 LabelEncoding을 진행해주세요.

(Caution) 인코딩 과정에서도, Test Data에 전처리를 진행할 때는 Train Set과 동일한 전처리를 진행해줘야 합니다. 즉, Test set의 정보가 반영되는 Data Leakage가 발생해서는 안 됩니다.

(HINT) sklearn.preprocessing 의 LabelEncoder() 또는 OrdinalEncoder()를 사용하면 범주형 자료들을 LabelEncoding할 수 있습니다. test 데이터에만 있는 범주형 클래스를 처리하기 위해서는 OrdinalEncoder에 있는 handle_unknown 파라미터를 조정하여 처리해야 Data Leakage가 발생하지 않습니다.

문제6. 랜덤포레스트 모델을 사용하겠습니다. 패키지로 해당 모델을 불러와주세요.

- from sklearn.ensemble import RandomForestRegressor

문제 6-1. 항상 같은 결과를 얻을 수 있도록 random_state 를 3233 으로 고정해주세요.

- RandomForestRegressor(random_state=3233)

문제 6-2. 모델을 train으로 학습시킨 후, test 데이터로 예측을 수행해주세요. 예측 결과와 answer.csv의 값을 비교하여 MSE(Mean Squared Error)를 계산해주세요.

문제7. 새롭게 범주형 피쳐 엔지니어링을 실시하겠습니다. 문제3까지 처리 완료한 데이터를 다시 train, test에 저장해주세요.

문제7-1. Target Encoding이 무엇인지 알아본 후, 간단하게 설명해주세요.

문제7-2. Cardinality(변수의 unique한 값의 개수)가 10 이상인 열에 대해 ('Comp', 'Color') Mean Target Encoding을 수행해주세요. 만약 train에 없는 범주형 클래스로 인해 test에 결측치가 발생한 경우 train data의 평균 값으로 대체해주세요.

문제7-2. Cardinality가 10 이하인 열에 대해서는 One-Hot Encoding을 수행해주세요.

(HINT1) sklearn.preprocessing의 OneHotEncoder()를 사용해주세요. pandas 라이브러리의 get_dummies의 경우 Data Leakage의 위험이 큽니다.

(HINT2) 원 핫 인코딩의 경우에도 마찬가지로, handle_unknown 파라미터를 조정하여 test data에만 있는 범주형 클래스를 적절하게 처리할 수 있습니다.

문제8. 인코딩이 완료된 train, test에 대해 문제2, 문제6을 수행해주세요. 결과를 비교해주세요.

문제8-1. Target Encoding이 해당 데이터에서 더 좋은 결과를 낼 수 있었던 이유를 데이터나 도메인적 측면에서 고민해본 후 간단하게 작성해주세요. 이러한 방식의 인코딩이 언제든 적절할지 생각해본 후 의견을 작성해주세요.

Chapter 2. Factor Analysis

요인분석은 변수 간 상관구조를 파악할 때 유용하게 사용할 수 있는 알고리즘입니다. 요인분석은 변수들 간 상관관계를 고려하여 내재된 요인을 추출한 뒤, 요인 별로 변수를 묶어주는 방법이라고 할 수 있습니다. 이러한 요인 분석은 해석이 중요한 분야, 또는 공모전에서 유용하게 활용되기도 합니다. 해당 분석 기법을 직접 수행해보며 데이터 해석 능력을 길러봅시다.

문제1. 요인분석과 PCA의 차이점이 무엇인지 살펴본 후, 간단하게 서술해주세요.

문제2. 챕터 1 - 문제7의 인코딩이 완료된 데이터를 사용합니다. 데이터에서 log_Price 열을 제거한 후, 변수들의 scale에 대한 영향을 제거하기 위해 스케일링을 진행해주세요.

(HINT) sklearn.preprocessing의 StandardScaler를 진행하면 편리합니다. 다만 이 방법을 사용할 경우, 변환된 데이터를 다시 데이터프레임으로 변환해주는 과정이 요구됩니다.

문제3. 스케일링된 데이터에 대해 5개의 factor로 요인분석을 실시해주세요.

(HINT) factor_analyzer의 FactorAnalyzer를 사용해주세요. rotation="varimax" 를 사용합니다.

문제4. Factor 1부터 Factor 5까지의 요인 적재량(Loading)을 확인하고, 각 Factor들이 어떤 요인을 의미하는지 추측하여 서술해주세요. (즉, 각 요인들에 적절한 이름을 붙여보세요. 정답은 없습니다.)

(HINT) fa(내가 붙인 모델 이름).loadings_를 데이터프레임화하면 좀 더 가독성 있게 만들 수 있습니다.

Chapter 3. eXplainable AI(XAI)

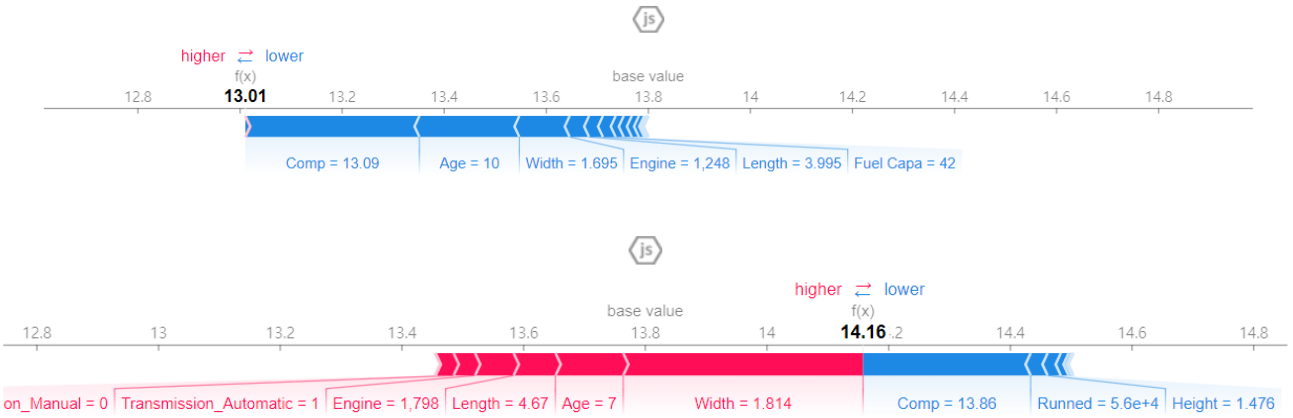
Neural Network나 랜덤 포레스트 모델은 높은 예측 성능을 가지지만, 그만큼 모델의 해석에는 어려움을 가진다는 단점이 있습니다. 이러한 모델의 설명력을 보충하기 위해 만들어진 대표적인 방법론으로 LIME과 SHAP가 있습니다. XAI를 통해 모델이 어떤 근거로 예측 결과를 낸 것인지 확인할 수 있다면 모델을 신뢰할 수 있을 뿐만 아니라 모델이 적절하였는지, 학습 시 사용한 데이터가 적절하였는지도 파악해볼 수 있을 것입니다.

이번 챕터에서는 간단한 예제를 구현해보며 XAI에 대해 경험해보는 시간을 갖도록 하겠습니다. 추가로 2023-2 데이터마이닝팀 클린업 3주차에 이론 설명, 시계열자료분석팀 주제분석 4주차에 SHAP를 활용한 실제 분석 사례가 있으니 참고해주시면 공부에 도움이 될 것 같습니다.

문제1. Shapely Value가 어떤 식으로 계산되는지 찾아보고, 이해한 바를 간단하게 적어주세요.

문제2. 챕터 1 - 문제7의 인코딩이 완료된 데이터를 사용합니다. 챕터 1에서 훈련시킨 랜덤 포레스트 모델에 대해 shap Explainer 객체를 만들어 주세요. 그리고 테스트 데이터셋(test)에 대해 shap_values를 계산해 변수에 저장해주세요.

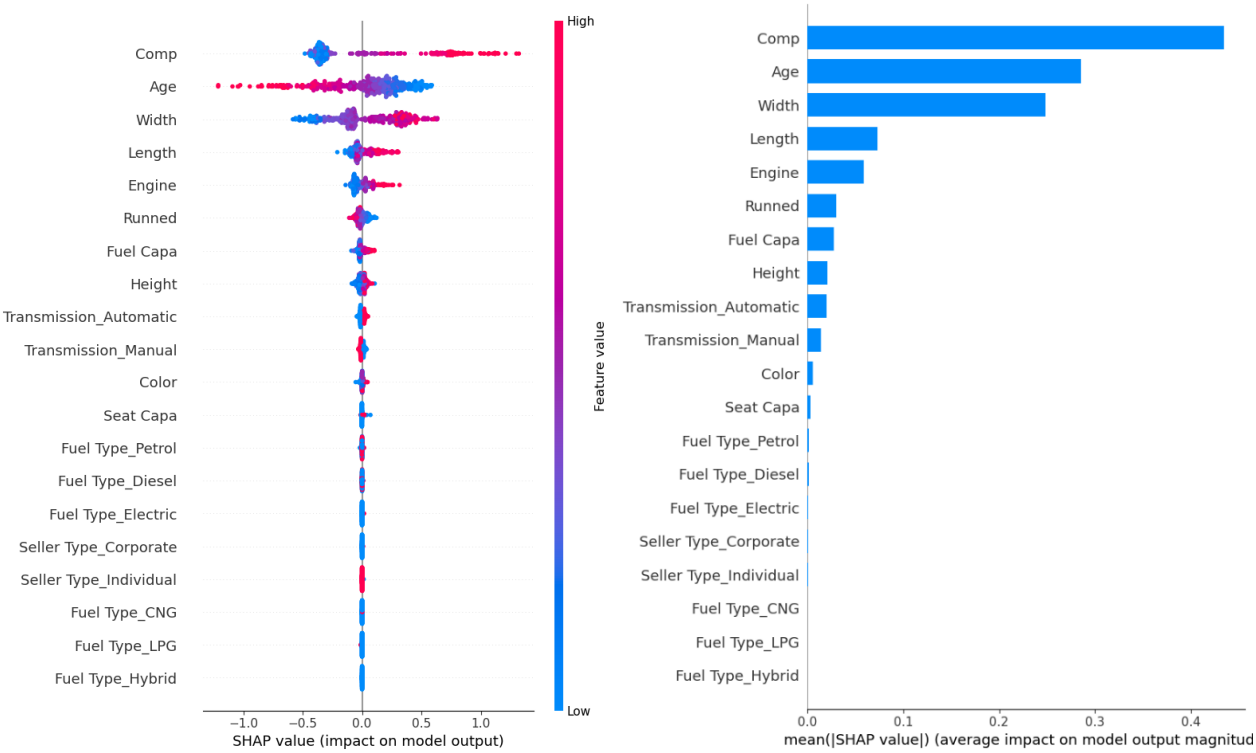
문제3. SHAP은 모델의 예측 하나 하나에 대해서 어떤 변수가 예측에 얼마나 영향을 주었는가를 알려줍니다. 첫 번째와 두 번째 테스트(test) 데이터에 대한 shap value를 force plot으로 나타내고 그래프를 해석해주세요.



(HINT) shap.force_plot()을 통해 위 plot을 쉽게 생성할 수 있습니다.

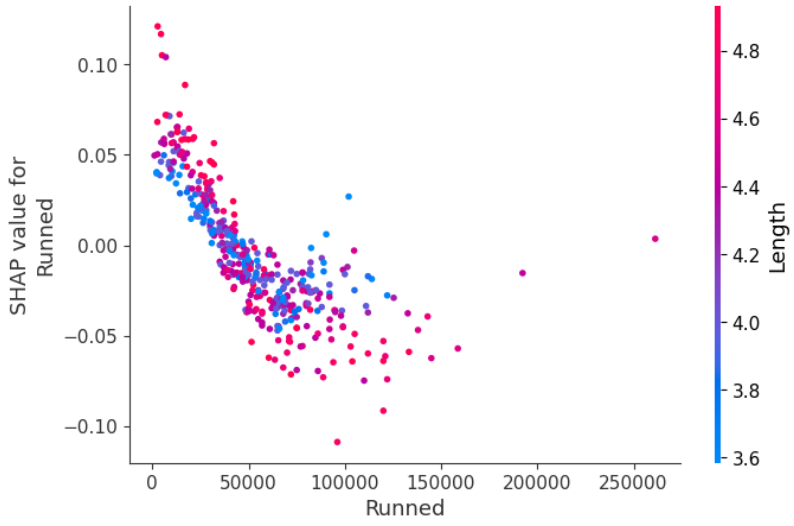
문제4. 문제 3은 하나 하나, 즉 각각의 데이터에 대한 변수들의 영향을 확인하는 작업이었습니다. 각 변수에 대한 Shap Values의 값들, 또 절댓값을 통해 변수들의 전반적인 영향력을 파악해주세요.

(HINT) shap.summary_plot() 으로 아래 plot들을 쉽게 생성할 수 있습니다. 또한 plot_type = "bar"로 설정하면 오른쪽 plot을 생성할 수 있습니다.



문제5. 위 분석 결과를 보고, log_Price와 음의 상관관계를 가지는 것으로 보이는 변수들을 파악해보세요.

(BONUS) Runned 변수에 대한 Dependence plot을 그려보세요. 결과를 간단하게 해석해주세요.



(HINT) shap.dependence_plot()을 사용하면 위 plot을 쉽게 생성할 수 있습니다.