Maven Roasters 카페 일별 매출 예측

이름: 김태수
학번: 202016219

데이터 크기

- 관측치 149,116개
- 변수 11개

사용된 변수

- transaction_id : 각 거래의 고유 ID (일련번호)
- transaction_date : 거래 날짜 (MM/DD/YY)
- transaction_qty : 거래된 수량 (몇 개 팔렸는지)
- store_id : 매장의 고유 ID (예: S1, S2 등)
- store_location : 매장 위치 (예: Manhattan, Brooklyn 등)
- product_id : 제품 고유 ID
- unit_price : 제품 1개의 가격
- product_category : 제품 카테고리 (예: Beverage, Food 등)
- product_type : 제품 종류 (예: Coffee, Tea, Sandwich 등)
- product_detail : 세부 제품 이름 (예: Iced Vanilla Latte 등)
- daily_sales : 일별 매출 (반응변수)

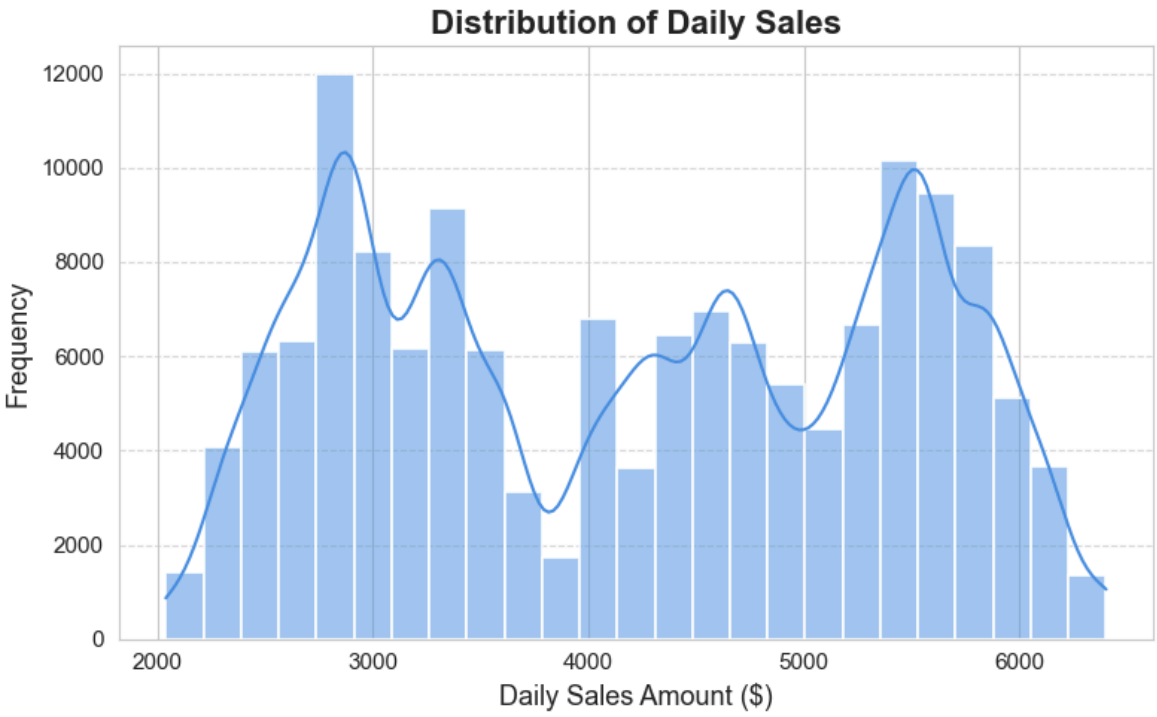
기본 데이터 구조

Out[7]:

	transaction_id	transaction_date	transaction_qty	store_id	store_location	product_id
0	1	2023-01-01	2	5	Lower Manhattan	32
1	2	2023-01-01	2	5	Lower Manhattan	57
2	3	2023-01-01	2	5	Lower Manhattan	59
3	4	2023-01-01	1	5	Lower Manhattan	22
4	5	2023-01-01	2	5	Lower Manhattan	57

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 149116 entries, 0 to 149115
Data columns (total 11 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   transaction_id    149116 non-null  int64
 1   transaction_date  149116 non-null  datetime64[ns]
 2   transaction_qty   149116 non-null  int64
 3   store_id          149116 non-null  int64
 4   store_location    149116 non-null  object
 5   product_id        149116 non-null  int64
 6   unit_price        149116 non-null  float64
 7   product_category  149116 non-null  object
 8   product_type      149116 non-null  object
 9   product_detail    149116 non-null  object
 10  daily_sales       149116 non-null  float64
dtypes: datetime64[ns](1), float64(2), int64(4), object(4)
memory usage: 12.5+ MB
None
```

반응변수 분포



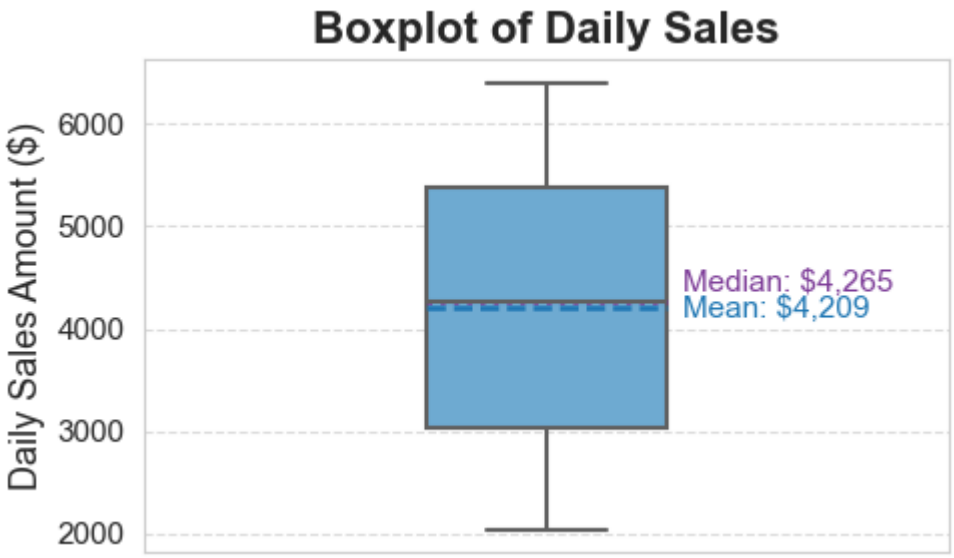
- daily_sales는 대체로 2000~6000달러 사이에 분포하며, 중앙값 부근인 약 4000달러 근처에서 빈도가 다소 감소하는 경향이 관찰된다.
- 이는 히스토그램 상에서도 중심부가 평탄하지 않고 살짝 패여 있는 형태로 나타난다.

기초통계량

Out[10]:
```
count    149116.000000
mean       4209.175173
std        1202.722643
min        2037.100000
25%        3040.250000
50%        4265.450000
75%        5370.810000
max        6403.910000
Name: daily_sales, dtype: float64
```

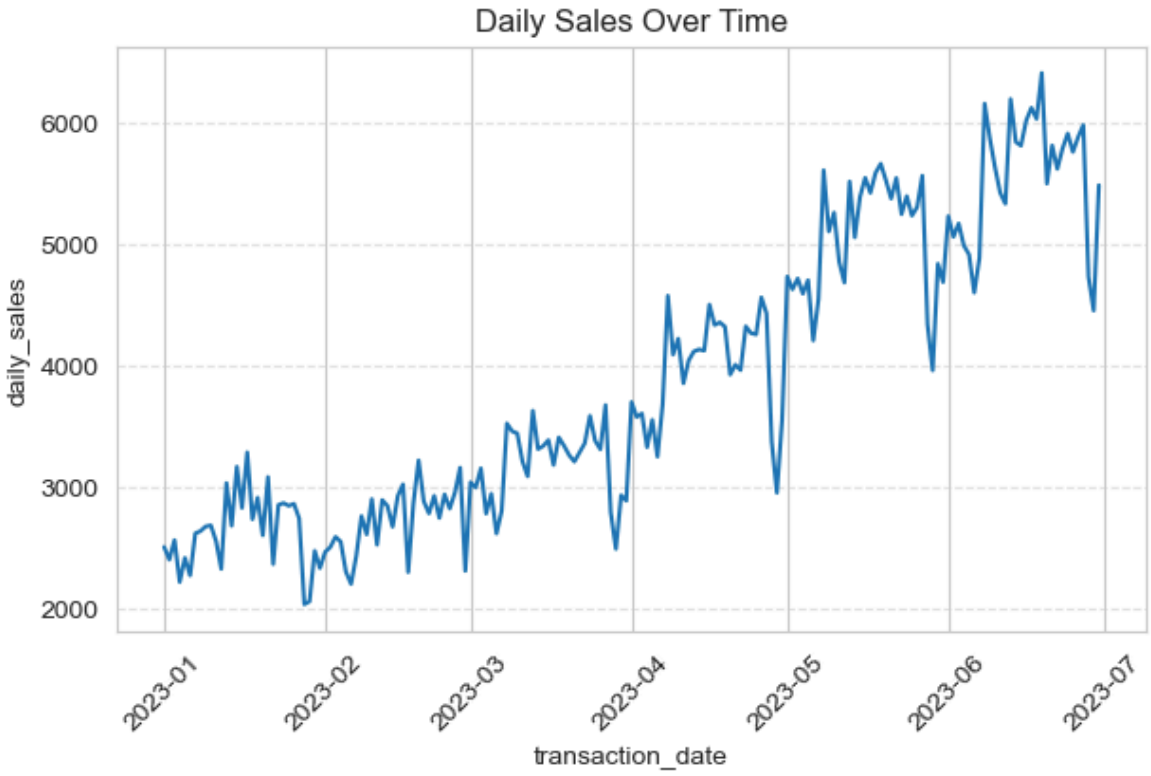
- daily_sales의 평균은 약 4209.18, 중앙값은 4265.45로 양자의 차이가 약 56.27에 불과하여 분포는 대체로 대칭적인 형태로 보인다.
- 또한 표준편차는 약 1202.72로, 평균 대비 약 28.6% 수준으로 분산도 과도하지 않아 데이터의 산포도 비교적 안정적인 편이다.

이상치



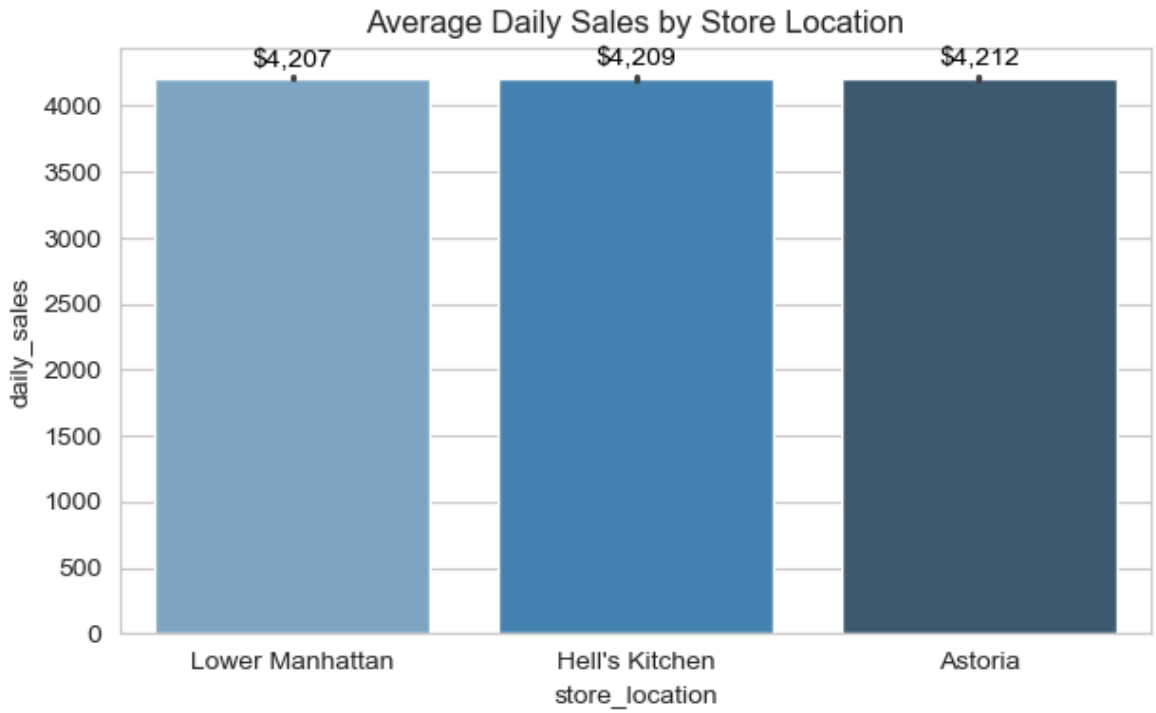
- 박스플롯을 활용한 IQR(Interquartile Range) 기반 탐지 결과, 대부분의 관측값이 1.5×IQR 범위 내에 분포하고 있으며, 극단적인 이상치는 발견되지 않았다.

일별 매출액



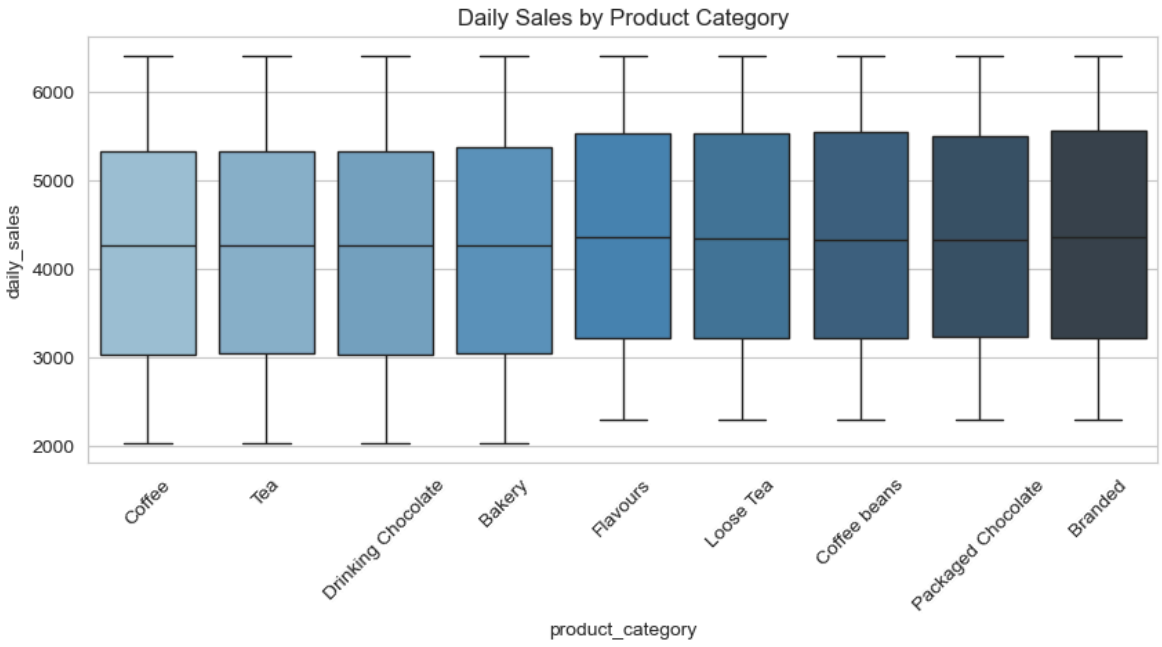
- 전체적으로 시간 경과에 따라 daily_sales가 점진적으로 증가하는 추세를 보인다.
- 월별 매출 흐름을 보면, 매월 초에는 매출이 상대적으로 증가하는 반면, 월말에는 점차 감소하는 경향이 반복적으로 나타나므로 월 단위 주기성을 가지는 것 같다.

지점별 매출액



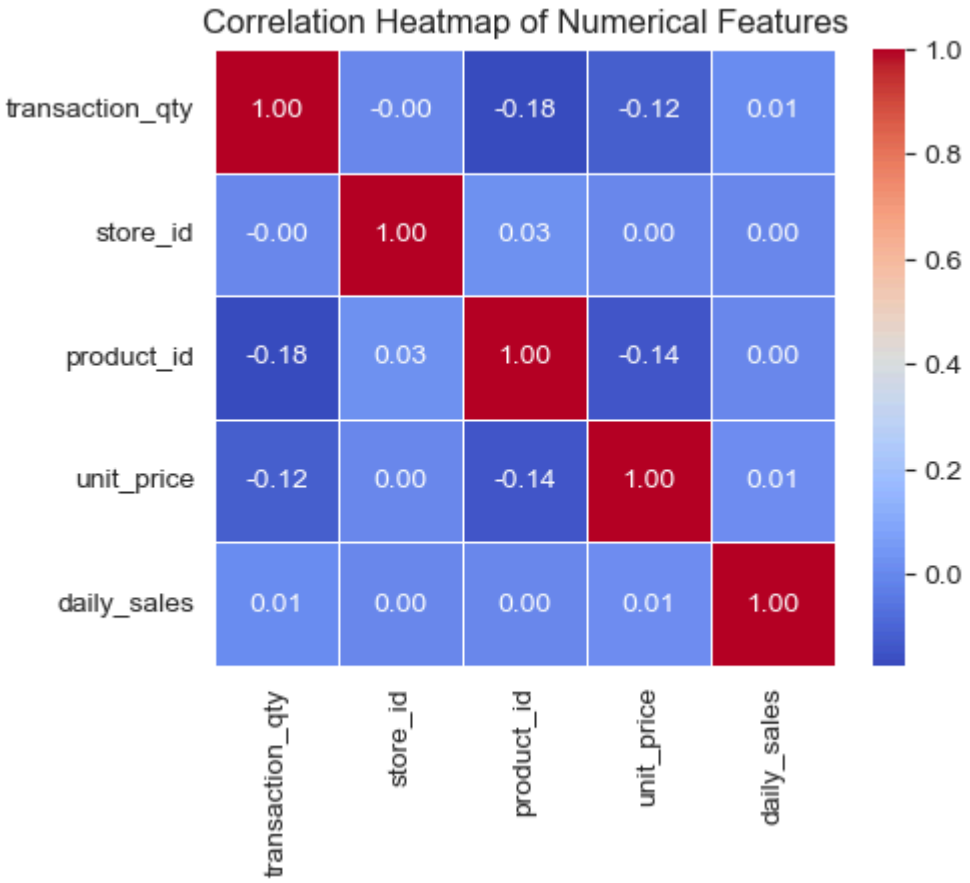
- 지점별 daily_sales 분포를 비교한 결과 전체적으로 유사한 범위내에 분포되어 있어 지점 간 매출 차이는 유의하다고 볼 수 없다.

제품 카테고리별 매출액



- product_category별로 daily_sales의 분포를 비교한 결과, 각 그룹 간 평균 매출액 차이가 미미하고 분산 또한 유사하게 나타난다.

상관계수 Heatmap



- 수치형 변수 간 상관관계를 살펴본 결과 반응변수인 daily_sales와 다른 변수들 간의 상관계수는 모두 0.1 이하로 매우 낮아 강한 선형적 연관성은 확인되지 않는다.

전처리

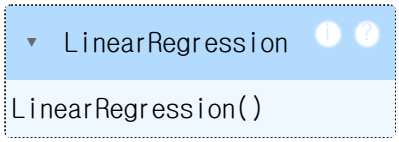
- transaction_date는 날짜형 변수로서 회귀모형에 직접 입력하기 적절하지 않기 때문에, 연(year), 일(day), 요일(dayofweek) 의 파생변수로 분해하여 사용하였다.
- transaction_id는 단순히 각 거래를 구분하기 위한 식별자 역할만 하므로 예측에 기여할 정보가 없어 분석에서 제외하였다.
- 또한 범주형 변수는 회귀모형에서 활용할 수 있도록 원핫인코딩 방식으로 변환하였다.

데이터 분할

- 데이터는 훈련용 70%, 평가용 30%로 분할하여 모델 학습 및 성능 평가에 활용하였다.

회귀모형

Out[23]:

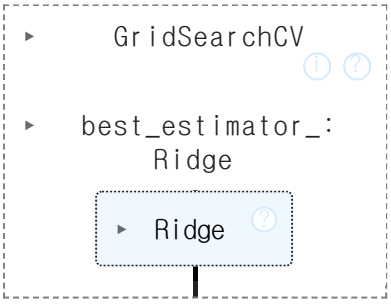



```
[Linear Regression] RMSE: 1194.48
```

- 선형 회귀는 종속 변수와 독립 변수 간의 선형 관계를 가정하는 가장 기본적인 예측 모델로, 모든 설명변수를 그대로 사용하여 학습하였다.
- 범주형 변수는 원핫인코딩 처리되었으며, 날짜 변수는 연, 일, 요일로 파생된 후 인코딩되었다.
- 데이터는 훈련 70%, 테스트 30%로 분할하여 학습하였고, 예측 성능 평가는 RMSE를 기준으로 진행하였다.
- 선형 회귀 결과, 테스트 데이터에 대한 RMSE는 1194.48이였다.
- 이는 선형 모델이 비선형 패턴을 충분히 포착하지 못했기 때문일 수 있으며, 이후 더 복잡한 모델에서 성능 향상을 기대할 수 있다.

RIDGE

Out[29]:



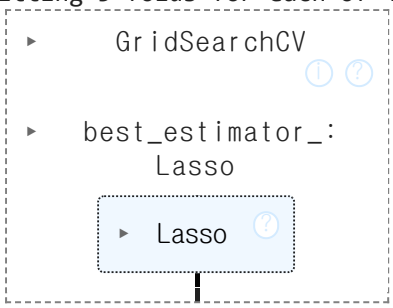

```
RMSE: 1194.48
```

- Ridge 회귀는 선형 회귀에 L2 정규화 항을 추가하여 계수의 크기를 제한함으로써 과적합을 방지하는 모델이다.
- 설명변수들은 StandardScaler를 사용하여 정규화하였고, GridSearchCV를 통해 alpha 값에 대해 5-fold 교차검증을 수행한 결과, 최적의 alpha는 100으로 선택되었다.
- 데이터는 훈련 70%, 테스트 30%로 분할하여 학습하였고, 예측 성능 평가는 RMSE를 기준으로 진행하였다.
- 테스트 데이터에 대한 RMSE는 1194.48으로, 일반 선형 회귀와 유사한 수준의 예측 성능을 나타냈다.

LASSO

```
Fitting 5 folds for each of 4 candidates, totalling 20 fits
```

Out[33]:




```
RMSE: 1194.39
```

- Lasso 회귀는 선형 회귀에 L1 정규화 항을 추가하여 일부 계수를 0으로 만들 수 있어, 자동 변수 선택 효과를 가지는 모델이다.
- 설명변수들은 StandardScaler를 사용하여 정규화하였고, GridSearchCV를 통해 alpha 값에 대해 5-fold 교차검증을 수행한 결과, 최적의 alpha는 1로 선택되었다.
- 데이터는 훈련 70%, 테스트 30%로 분할하여 학습하였고, 예측 성능 평가는 RMSE를 기준으로 진행하였다.
- 테스트 데이터에 대한 RMSE는 1194.39로, 선형 회귀 및 Ridge 회귀와 유사한 수준의 예측 성능을 나타냈다.

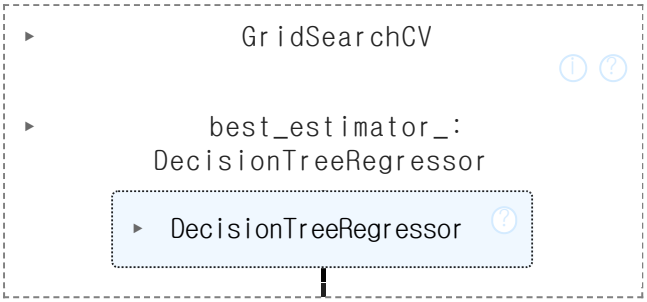
GAM

```
[GAM] RMSE: 1165.19
```

- GAM(Generalized Additive Model)은 선형 회귀의 확장된 형태로, 각 설명변수에 대해 개별적으로 비선형 함수를 적용함으로써 더 유연한 예측이 가능하도록 한다.
- pygam 라이브러리를 사용하여 LinearGAM 모델을 구성하였으며, 기존 전처리된 설명변수를 그대로 입력하여 학습을 수행하였다.
- 데이터는 훈련 70%, 테스트 30%로 분할하여 학습하였고, 예측 성능 평가는 RMSE를 기준으로 진행하였다.
- 테스트 데이터에 대한 RMSE는 1165.19으로, 기존 모델들보다 약간 향상된 예측 성능을 나타냈다.

나무모형

Out[42]:

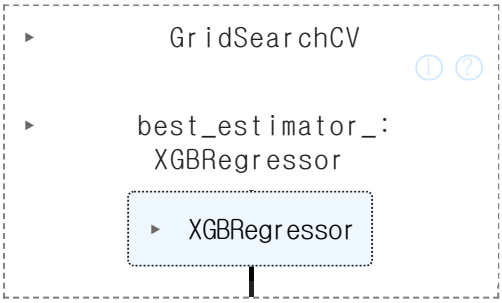



```
RMSE: 137.57
```

- 의사결정나무는 데이터를 조건에 따라 분할하면서 예측값을 생성하는 비선형 모델로, 변수 간 상호작용과 비선형 구조를 자동으로 포착할 수 있는 장점이 있다.
- GridSearchCV를 이용해 max_depth와 min_samples_split을 최적화한 결과, 최적 파라미터는 max_depth=15, min_samples_split=10으로 선택되었다.
- 데이터는 훈련 70%, 테스트 30%로 분할하여 학습하였고, 예측 성능 평가는 RMSE를 기준으로 진행하였다.
- 테스트 데이터에 대한 RMSE는 137.57로, 기존 모델들보다 현저히 향상된 예측 성능을 나타냈다.

부스팅

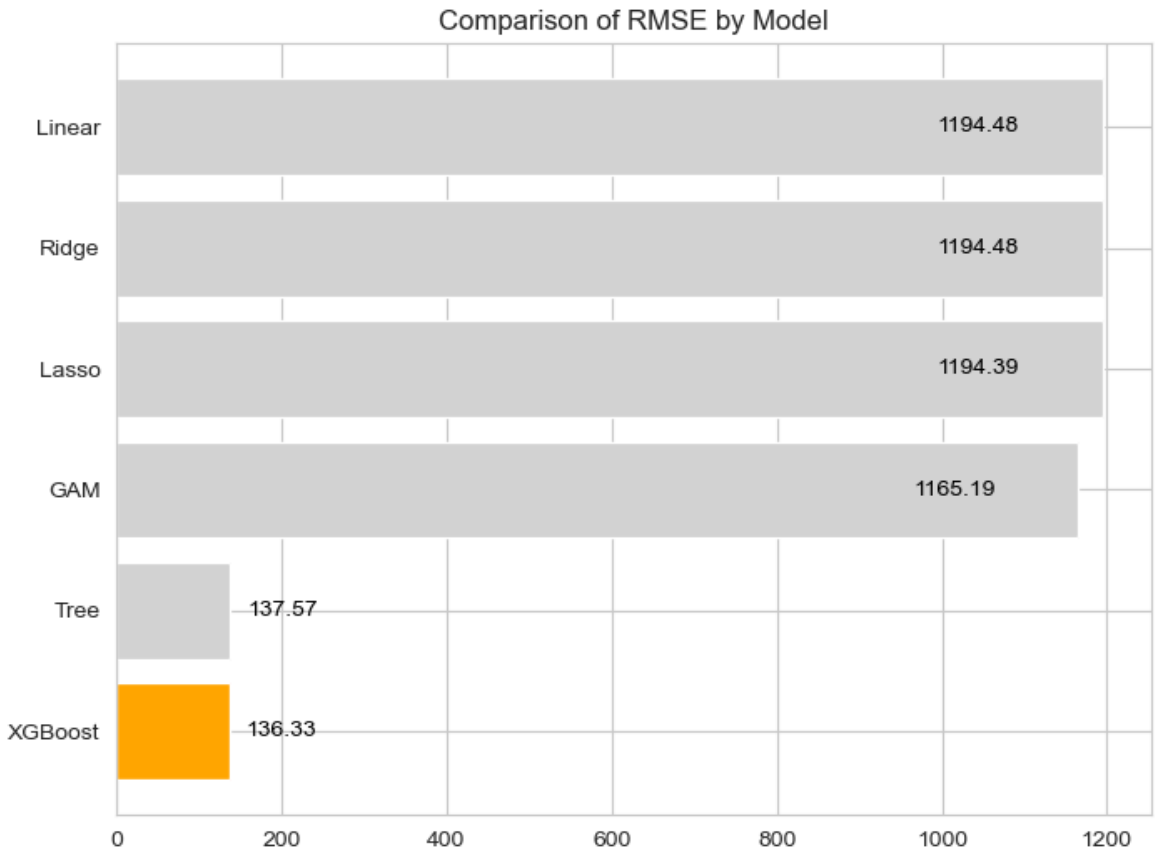
Out[48]:




```
RMSE: 136.33
```

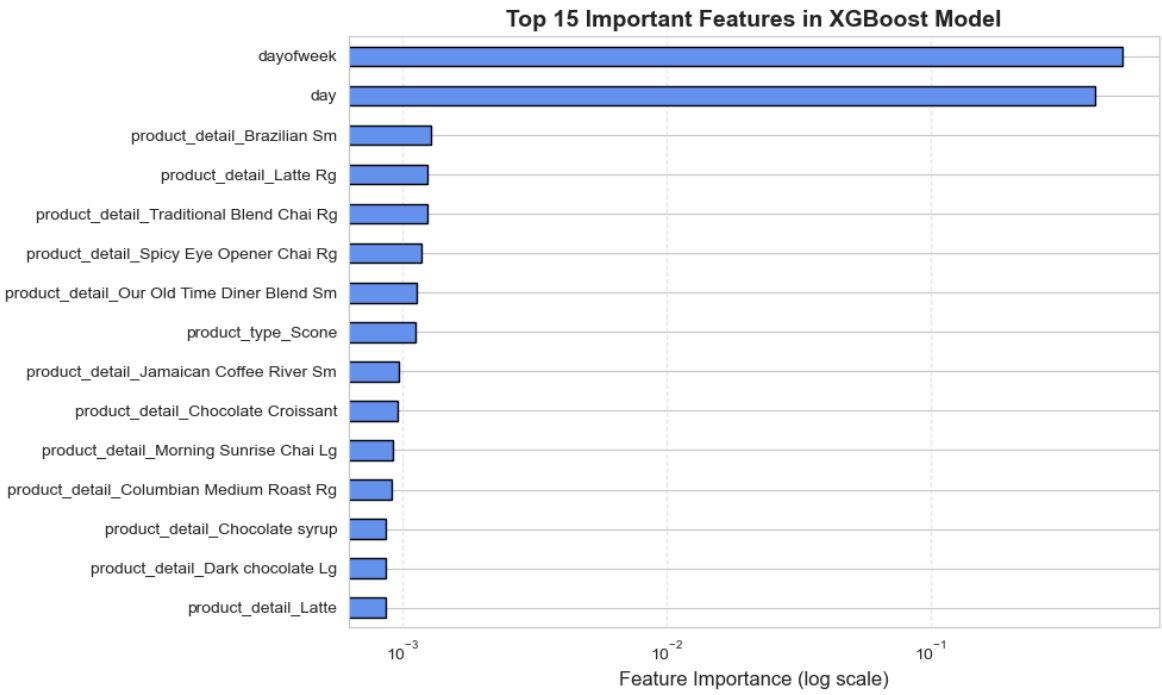
- XGBoost 모형은 여러 약한 결정나무를 순차적으로 학습하며 잔차를 보정하는 부스팅 앙상블 방식으로, 복잡한 변수 간 상호작용과 비선형성을 효과적으로 포착한다.
- GridSearchCV를 통해 max_depth=7, learning_rate=0.2, n_estimators=200, min_child_weight=5의 최적 파라미터를 도출하였다.
- 데이터는 훈련 70%, 테스트 30%로 분할하여 학습하였고, 예측 성능 평가는 RMSE를 기준으로 진행하였다.
- 테스트 데이터 RMSE는 136.33으로 의사결정나무보다 성능이 약간 향상되었다.

성능비교



- 선형모형 계열에서는 모두 RMSE가 1194 내외로 유사한 수준이었으나, 이는 데이터가 선형모형으로는 잘 설명되지 않음을 의미한다.
- 반면, 트리 기반 모델은 변수 간 상호작용 및 비선형 분포를 효과적으로 포착함으로써 RMSE를 137 수준까지 낮추었고, XGBoost는 여기에 더해 오차 보정을 통해 가장 우수한 성능을 기록하였다.

변수 중요도



- 최적의 모형으로 뽑힌 XGBoost에서 어떠한 변수가 크게 중요한지 살펴보았을 때 거래 요일에 해당하는 dayofweek와 거래일에 해당하는 day의 중요도가 가장 높게 나왔다.
- 이는 특정 요일이나 날짜에 따라 매출액이 크게 달라지며 고객 행동 패턴(예: 주말에 더 붐빔, 평일은 한산함 등)을 반영한 결과로 받아들일 수 있다.