호텔 방문자의 예약 취소 예측 보고서

2조

12223578 김수민
12223590 백소영
12211262 우지민
12223614 추민경

< 목차 >

1. 서론

- **1.1 분석 목표**
- **1.2 가설 설정**

2. 본론

- **2.1 데이터 셋 설명**
- **2.2 고객 특징 분석**
- **2.3 Pre Processing**
- **2.4 Modeling**
- **2.5 Model Comparison**
- **2.6 교차 검증**
- **2.7 고객 유형 시각화**

3. 결론

- **3.1 가설확인**
- **3.2 의사결정 방향**
- **3.3 개선 방안**

1. 서론

온라인으로 시설이나 서비스를 이용하기 위해 예약시스템에 접속하면 흔하게 볼 수 있는 것이 바로 예약금 제도이다. 예약금 제도란 시설 혹은 서비스를 이용하기 위해 예약한 고객의 노쇼(No show)를 방지하기 위해 도입된 제도로 전체 이용금의 일부 혹은 가게마다 지정한 금액을 이용하기 전에 미리 지불하는 제도이다. 예약금 제도가 도입된 것은 예약을 하고 시설이나 서비스를 이용하지 않는 고객의 비율이 증가했기 때문이다.

예약금 제도는 특정 업종만이 아니라 다양한 업종에서 실시하고 있는데 그 중 호텔의 경우 예약한 고객이 노쇼를 하는 경우 당일 새로운 고객을 받는 것이 다른 업종에 비해 어렵고 이용요금이 크기 때문에 그만큼 호텔이 손해를 보는 금액이 크다.

호텔에서도 고객들의 갑작스러운 취소 또는 노쇼를 방지하기 위해 보증금 제도 혹은 취소 시에 수수료는 부과하는 등의 제도를 실시하고 있다. 추가적으로 '오버부킹(Over booking)'을 시행하고 있는 호텔도 있지만 이는 피해가 고객에서 돌아갈 가능성이 높기 때문에 오버부킹 제도를 실시하는 것이 쉽지 않다.

1.1 분석 목표

본 분석의 목표는 **'Hotel Booking Prediction' 데이터셋을 이용하여 고객의 예약 취소를 예측**하여 기업의 손실을 줄이고자 한다. 고객의 예약취소 혹은 노쇼로 인한 호텔의 손실을 줄이는 것에 있어서 다양한 제도의 도입이 도움을 줄 수 있지만 여전히 어느 정도의 손실을 감수해야 하거나 고객에게 피해가 갈 수 있다는 위험부담이 존재한다. 따라서 고객의 예약취소를 미리 예측할 수 있다면 호텔의 손실도 줄이고 고객에게 돌아가는 피해 또한 줄일 수 있을 것이다.

① 호텔 예약 고객층 분석
② 호텔 예약 취소 여부에 대한 예측 및 중요 변수 알아보기

1.2 가설설정

1.2.1 고객 유형

① 호텔 가격
: 휴가철인 7-8월에 가격이 높을 것이고 나머지 달의 가격은 상대적으로 낮을 것으로 예상했다.
② 방문 고객
: city hotel보다는 저렴한 resort hotel의 방문고객이 전반적으로 많을 것이다. 하지만 두 호텔의 방문 고객 경향성은 휴가철에 높고, 나머지는 낮은 형태로 비슷할 것으로 예상했다.
③ 숙박 기간
: 대부분의 사람들이 1박2일의 휴가를 보내기 위해서는 resort hotel을 찾을 것이고, 장기 숙박 고객은 city hotel에 오래 머물 것이다.

1.2.2 모델링

① 중요 변수
: 고객의 hotel booking cancel에 가장 많이 영향을 미치는 변수로 previous_cancellations(이전 취소 건수)를 예상하였다.
② 적합 모델

: 예약 취소를 예측하기 위한 모델로 Random Forest와 XGBoost를 선택하였다. 예약취소가 아닌 샘플의 수는 74745개, 예약취소 샘플 수는 44157개 이므로 데이터에 불균형이 있다는 판단을 하였다. Random Forest의 경우, 각기 다른 부트스트랩 샘플(데이터의 무작위 부분집합)을 사용해 다수의 트리를 구성하므로, 전체적인 데이터 구조를 보다 잘 반영할 수 있어 불균형 데이터셋에서도 비교적 안정적인 성능을 보이기에 Random Forest가 더 높은 성능을 보일 것으로 예상하였다.

2. 본론

2.1 데이터 설명

Kaggle의 “Hotel Booking Prediction” 데이터셋

2.1.1 columns

- hotel: 호텔 ('Resort Hotel' or 'City Hotel')
- is_canceled: 예약 취소 여부 (0: 취소되지 않음, 1: 취소됨)
- lead_time: 예약 시점과 도착 시점 사이의 일수
- arrival_date_year: 도착 연도
- arrival_date_month: 도착 월
- arrival_date_week_number: 도착 주차
- arrival_date_day_of_month: 도착 일
- stays_in_weekend_nights: 주말 밤 숙박 수
- stays_in_week_nights: 주중 밤 숙박 수
- adults: 성인 수
- children: 어린이 수
- babies: 유아 수
- meal: 식사 유형 (예: 'BB', 'HB', 'FB')
- country: 거주 국가
- market_segment: 시장 세분화 (예: 'Direct', 'Corporate')
- distribution_channel: 유통 채널 (예: 'TA/TO', 'Direct')
- is_repeated_guest: 반복 고객 여부 (0: 처음 방문, 1: 재방문)
- previous_cancellations: 이전 취소 건수
- previous_bookings_not_canceled: 이전에 취소되지 않은 예약 건수
- reserved_room_type: 예약된 객실 유형
- assigned_room_type: 배정된 객실 유형
- booking_changes: 예약 변경 횟수
- deposit_type: 보증금 유형 (예: 'No Deposit', 'Refundable', 'Non Refund')
- agent: 예약 에이전트 ID
- company: 회사 ID
- days_in_waiting_list: 대기 리스트에 있었던 일수
- customer_type: 고객 유형 (예: 'Transient', 'Contract')
- adr: 평균 일일 요금
- required_car_parking_spaces: 필요한 주차 공간 수

- total_of_special_requests: 총 특별 요청 수
- reservation_status: 예약 상태 (예: 'Check-Out', 'Canceled')
- reservation_status_date: 예약 상태 날짜

2.1.2 Pre Processing 1

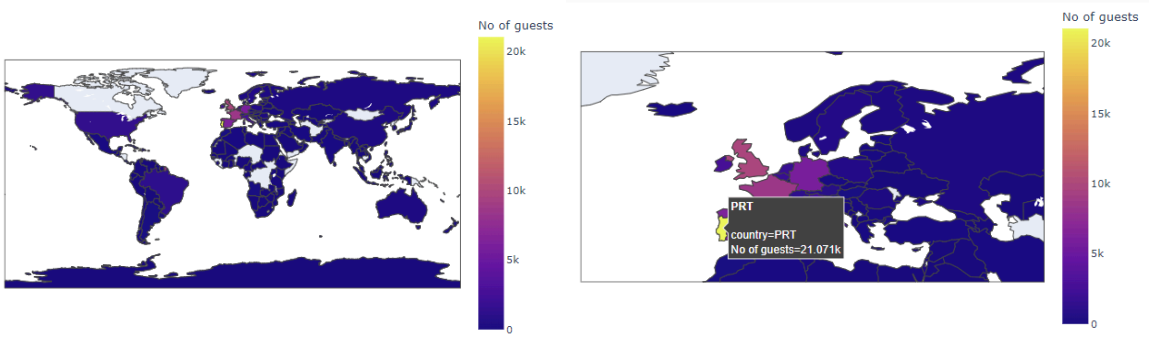
결측치가 있는 행을 제거하였다. ‘country’ 행의 경우, NaN값을 가진 행을 모두 삭제하였고, ‘agent’, ‘company’, ‘children’ 행의 경우, NaN값을 모두 0으로 대체하였다.

2.2 고객 특징 분석

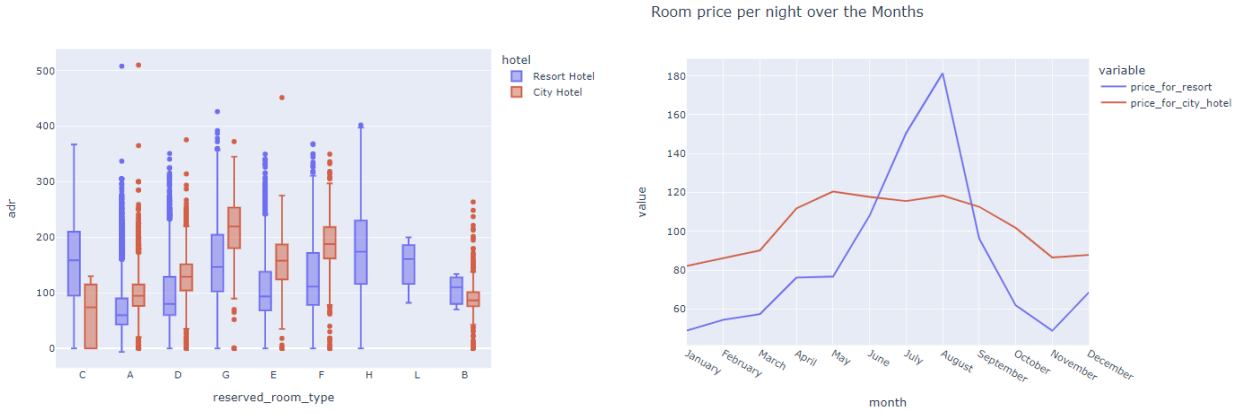
2.2.1 분석 결과

예약자들의 국가 분포(1), 선호하는 hotel 유형 및 가격(2), 호텔 가격 변동(3), 방문 고객 수(4), 머무는 시간(5)으로 분석을 진행하였다.

국가 분포에서는 PRT(21071) > GBR(9676) > FRA(8481)> ESP(6391)> DEU(6069) 순으로 나타났으며, 다양한 국적의 사람들이 resort 및 city 호텔에 머물지만 특히 포르투갈과 유럽권 사람들이 많이 방문함을 알 수 있었다. (1)

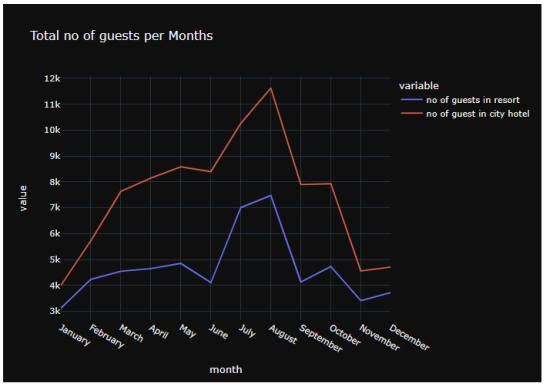


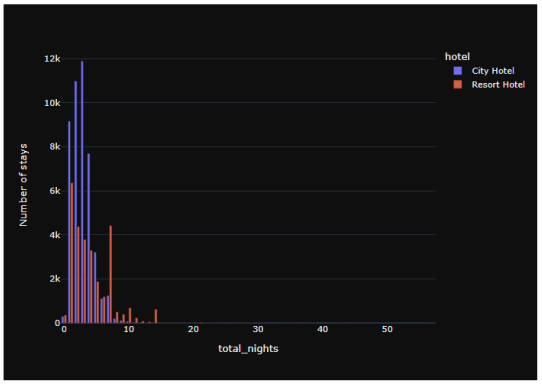
두번째로 선호하는 hotel 유형 및 가격에서는 선호하는 호텔 유형은 다양하게 분포되어 있음을 알 수 있었고 호텔의 가격 또한 광범위한 것을 알 수 있었다. (2)
세번째 호텔 가격 변동에서는 resort hotel은 변동이 크고 여름에 특히 가격이 높았고 이는 고객들이 여름 휴가철 성수기에 많이 찾아 가격이 높아진 것으로 예상하였다. city hotel은 여름을 제외한 모든 계절에 상대적으로 resort 보다 높은 일정한 가격을 유지하였으며 city hotel이 resort hotel 보다는 고급진 호텔이라 높은 가격을 유지하였다고 예상하였다. (3)



방문 고객 수는 전반적으로 city 보다는 resort hotel 방문자가 많은 것을 확인할 수 있었고 두 hotel 모두 여름철에

방문객이 증가하고 겨울철에 방문객이 감소함을 확인할 수 있었다. (4)
머무는 시간을 비교하면 두 hotel 모두 2~4일 머무는 고객이 제일 많았다. 하지만 resort hotel은 7일 머무는 고객이 특히 많은 것을 알 수 있었다.(5)





2.2.2 가설 확인

① 호텔 가격
: 휴가철인 7-8월에 가격이 높을 것이고 나머지 달의 가격은 상대적으로 낮을 것이다.
→ resort hotel은 여름철 가격 변동이 특히 큰 것을 확인할 수 있고 city hotel의 경우 가격 변동이 크지는 않지만 대체적으로 여름철 가격이 높은 것을 통해 *가설이 맞다고 할 수 있다.*
② 방문 고객
: city hotel보다는 저렴한 resort hotel의 방문고객이 전반적으로 많을 것이다. 하지만 두 호텔의 방문 고객 경향성은 휴가철에 높고, 나머지는 낮은 형태로 비슷할 것이다.
→ 전반적으로 city보다는 resort hotel 방문자가 압도적으로 많았고, 두 hotel 모두 여름철에 방문객이 증가하고 겨울철에 방문객이 감소하는 경향을 확인한 결과 *가설이 맞다고 할 수 있다.*
③ 숙박 기간
: 대부분의 사람들이 1박2일의 휴가를 보내기 위해서는 resort hotel을 찾을 것이고, 장기 숙박 고객은 city hotel에 오래 머물 것이다.
→ 총 방문객은 city hotel이 많았지만 장기 숙박 고객은 city보다 resort hotel에서 더 많았던 것을 확인한 결과, *가설이 틀렸다고 할 수 있다.*

2.3 Pre Processing 2

2.3.1 변수 선택

적절한 변수 선정을 위해 아래와 같은 이유로 몇 개의 변수를 제거하였다.

- 고객이 도착을 했다는 것은 cancel하지 않고 호텔에 방문했다는 것을 뜻하므로, 도착과 관련한 변수는 cancel과 상관이 없는 변수이기 때문에 arrival_date_year(도착 연도), arrival_date_month(도착 월), arrival_date_week_number(도착 주차) arrival_date_day_of_month(도착 일)은 제거하였다.

- agent(예약 에이전트 ID), company(회사 ID) 은 임의의 숫자로 표기되어있는 것을 확인했고, 이는 분류가 되어도 어떤 의미인지 알 수가 없다고 판단하여 제거하였다.

- assigned_room_type(배정된 객실 유형)은 방문 이후 배정받은 객실로 예약 취소 이유에 해당하지 않고, cancel 여부를 예측할 것이므로 reservation_status를 제거하였다.

- 범주형 자료인 country는 너무 광범위하여 다루기 어렵고, 다른 변수의 중요도가 더 높을것이라 예상하여 제거하였다.

2.3.2 범주형 변수 전처리

범주형 변수 column 'hotel', 'meal', 'market_segment', 'distribution_channel', 'reserved_room_type', 'deposit_type', 'customer_type', 'reservation_status_date' 에 대해서는 one-hot encoding을 진행하였다.

'reservation_status_date' 열에서 year과 month만 남기고 one-hot encoding을 진행하였다. (3.3 개선방안에서 추가 설명)

2.3.3 수치형 변수 전처리

수치형 변수 'lead_time', 'days_in_waiting_list', 'adr' 의 경우 분산이 매우 크지만 이후 사용할 모델에서는 정규화를 하지 않아도 되기 때문에 따로 진행하지 않았다. 또한 'adr' 열에 있는 모든 null 값을 평균값으로 대체하였다.

2.4 Modeling

여러 개의 모델를 생성하고 그 예측을 결합함으로써 보다 정확한 예측을 도출할 수 있고, 모델의 다양성이 증가함에 따라 예측의 안전성이 높아져 앙상블 기법을 선택하게 되었다.

① 분석 목표
: **호텔 예약 취소 여부에 대한 예측과 중요 변수를 알아보는 것이 목표**이기 때문에 예측이 가능해야 하고, 중요 변수를 추출할 수 있는 모델을 선택하였다. (1.1)

② 데이터셋의 비선형성
: 분석에 활용한 데이터 셋은 **자료들 간의 관계가 1:n의 관계**를 가지기 때문에 **비선형 자료구조** 중 트리구조를 선택하였다.

③ 고차원 데이터셋
: **columns의 개수가 60개가 넘는 고차원 데이터**으로 고차원에 용이한 모델을 선택하였다. (2.1)

④ 이상치에 둔감
: pre processing 과정에서 이상치 제거를 하지 않았고, **이상치에 강하다**. (2.1.2)

⑤ 효율성
: 각 상위 노드에 하위 노드가 여러개 존재하는 **계층 구조로 데이터를 저장**할 수 있다는 점, **데이터를 효율적으로 삽입, 삭제 및 검색**할 수 있다는 점과 **데이터 베이스 인덱싱을 통해 조회를 빠르게** 할 수 있는 모델을 선택하였다.

⑥ 변수 스케일에 둔감
: 다른 모델 (SVM, Logistic Reg, KNN)의 경우에는 표준화(정규화)가 필요하지만 **트리기반 모델(Decision Tree, Random Forest, AdaBoost, Gradient Boosting Classifier)은 변수의 스케일에 민감하지 않다**.(2.3.3)

2.4.1 Random Forest

Random Forest는 서로 다른 방향으로 과적합된 나무모형을 많이 만들어 이를 통합하여 과적합 정도를 감소하고 예측력은 유지한다. 회귀와 분류 문제에 모두 적용이 가능하며, 결측치 처리가 비교적 용이하여 Random Forest를 선택하게 되었다.

2.4.2 XGBoost

XGBoost는 L1(Lasso) 및 L2(Ridge) 정규화와 가중치 지정 가능이 포함되어 있어 모델의 복잡성을 제어하고 트리의 성장 과정에서 가지치기를 수행하여 과적합을 방지하며 최적의 트리 구조를 찾는다.

병렬 처리를 지원하므로 대용량 데이터셋에서도 빠른 학습과 예측이 가능하고,
각 특성의 중요도를 추정할 수 있으며 어떤 특성이 예측에 가장 큰 영향을 미치는지 파악하는 데 도움을 주어 XGBoost를 선택하게 되었다.

```
Accuracy Score of Random Forest is : 0.902905681005845
Confusion Matrix :
[[14325   701]
 [ 1608  7147]]
Classification Report :
              precision    recall  f1-score   support

           0       0.90      0.95      0.93     15026
           1       0.91      0.82      0.86      8755

    accuracy                           0.90     23781
   macro avg       0.90      0.88      0.89     23781
weighted avg       0.90      0.90      0.90     23781
```

```
Accuracy Score of Ada Boost Classifier is : 0.8650182919137126
Confusion Matrix :
[[14303   723]
 [ 2487  6268]]
Classification Report :
              precision    recall  f1-score   support

           0       0.85      0.95      0.90     15026
           1       0.90      0.72      0.80      8755

    accuracy                           0.87     23781
   macro avg       0.87      0.83      0.85     23781
weighted avg       0.87      0.87      0.86     23781
```

< Random Forest(좌) 결과 및 XGBoost(우) 결과 >

2.5 Model Comparison

아래와 같은 이유로 “Random Forest”와 “XgBoost” 중 최종 모델로 Random Forest를 선택하였다.

① 모델 성능

: 두 모델의 accuracy, f1-score,ROC-AUC를 비교하여 **Random Forest의 성능이 더 높은 것**을 확인하였다.

- f1-score은 양성 클래스 즉, 예약 취소의 예측 정확성을 파악할 때 유용하다.

- ROC-AUC은 모델의 전반적인 성능을 판단할 때 유용하다.

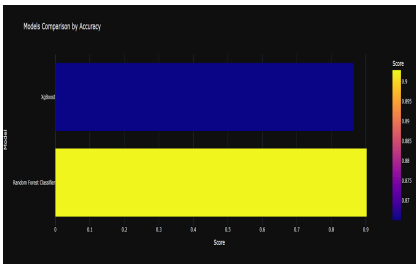
② 빠른 훈련 시간

: 그라디언트 부스팅 과정을 통해 점진적으로 모델을 개선해 나가는 XGBoost에 비해 **Random Forest는 병렬 처리가 가능하기 때문에 훈련 시간이 더 빠르**다. 시간제약이 있는 팀프로젝트 상황에 맞는 모델로 Random Forest를 선택하였다.

③ 이상치에 대한 robust

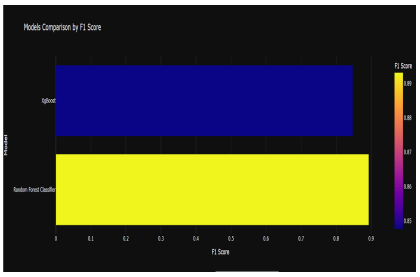
: Random Forest는 **여러개의 결정 트리를 사용하고 이들의 평균을 계산하므로 이상치에 상대적으로 덜 민감**하다. 수치형 데이터에 관한 columns만 확인해본 결과, 이상치가 많은 것을 확인할 수 있었다.

accuracy



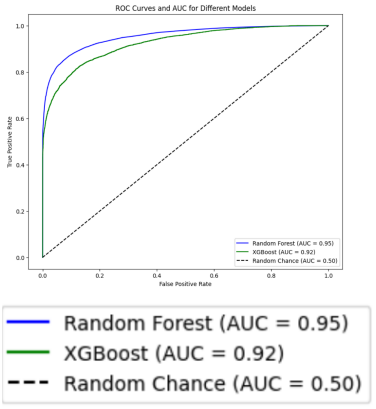
	Model	Score
0	Random Forest Classifier	0.902906
1	XgBoost	0.865018

F1-Score



	Model	F1-Score
0	Random Forest Classifier	0.893173
1	XgBoost	0.847623

ROC Curve



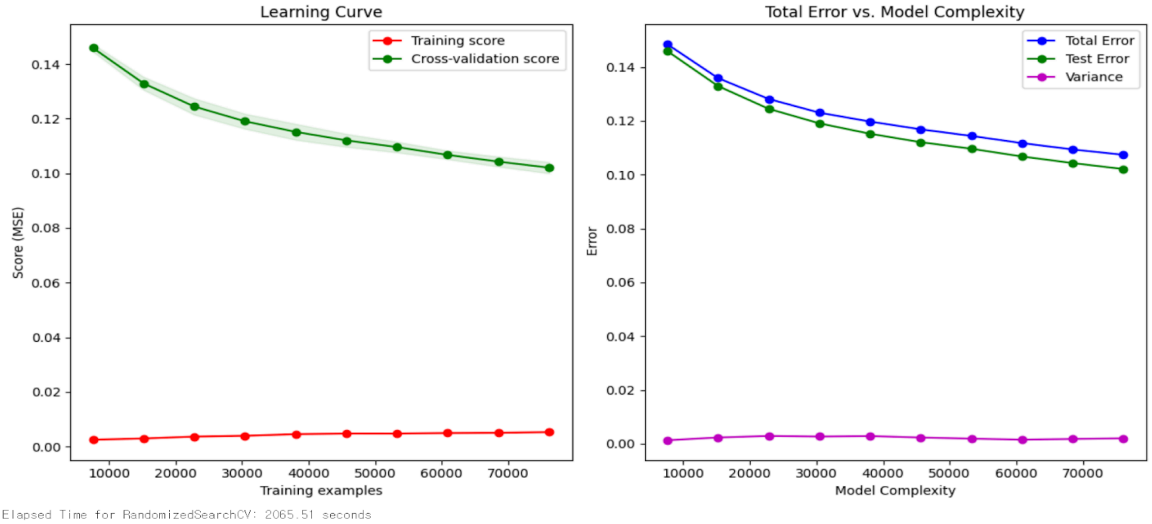
2.6 교차검증

하이퍼파라미터를 조정하여 최적의 조합을 찾기 위해 RandomizedSearchCV를 수행하였다. **분류기 개수는 100~500 범위에서 100단위로 조절하였고 'criterion'는 'gini'와 'entropy'로 설정**하여 교차검정을 진행하였다. 개별 트리의 랜덤성을 위해 트리의 깊이를 줄이는 하이퍼파라미터는 별도로 설정하지 않았다. 그 결과, 최적의 하이퍼파라미터 조합은 {'n_estimators': 400, 'criterion': 'gini'} 이고, 정확도는 약 0.903으로 우수한 성능을 보였다.

그리고 **learning curve와 분산-편향 그래프를 확인**하였다.

왼쪽 아래 그래프에서 학습 데이터가 매우 커졌음에도 불구하고 **Learning curve에서 기울기가 변하는 지점이 발생하지 않는 것**은 '모델 복잡성의 한계'로 Random Forest의 경우에 **모든 트리에 대해 과적합**시켜 평균을 내는 방법이기 때문에 과적합 성능을 개선할 수가 없어 위와 같은 양상을 보일 수 있다. Cross-validation Score의 우하향 경향 또한 모델이 과적합되고 있다는 것을 의미한다.

오른쪽 아래의 분산-편향 그래프를 통해 모델 복잡성이 증가함에 따라 편향의 경우, 지속적으로 감소하고 있고, 분산의 경우에는 일정하게 유지되고 있음을 확인할 수 있었다. **총 오류는 모델 복잡성에 따라 점차 감소하는 경향**을 보이며 일반화가 잘 이루어지고 있음을 보이지만 변화가 완만해지는 지점에서는 오버피팅이 발생할 수 있으므로 **적절한 지점을 찾아야** 한다.



< Learning Curve(좌)와 분산-편향 그래프(우) >

2.7 고객 유형 시각화

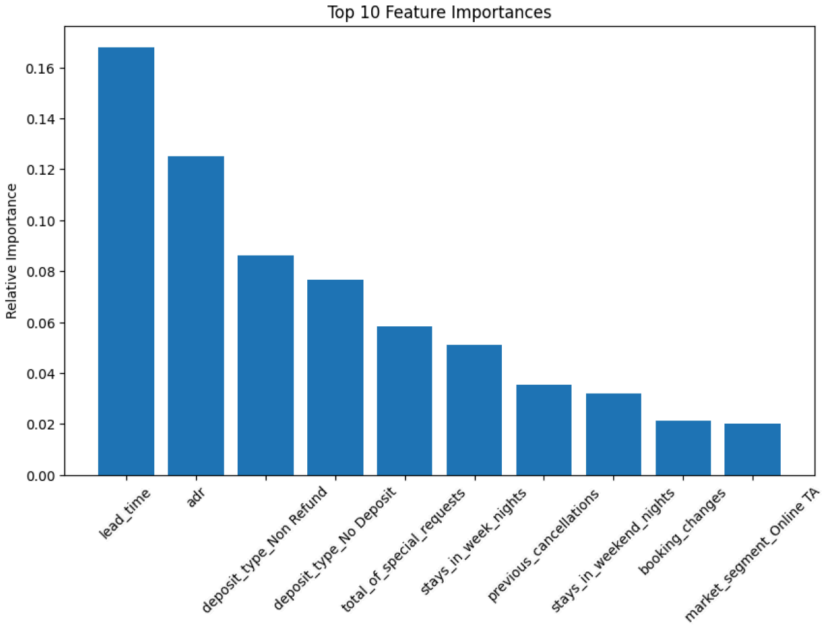
2.7.1 Feature Importance

앞서 RandomizedSearchCV를 통해 구한 최적모델의 특성 중요도를 확인하였다.

- lead_time: 예약 시점과 도착 시점 사이의 일수
- adr: 평균 일일 요금
- deposit_type_Non_Refund: 보증금 유형 중 "환불 불가"
- deposit_type_No_Deposit: 보증금 유형 중 "보증금 없음"
- total_of_special_requests: 총 특별 요청 수
- stays_in_week_nights: 주중 밤 숙박 수
- previous_cancellations: 이전 취소 건수
- stays_in_weekend_nights : 주말 밤 숙박 수
- booking_changes : 예약 변경 횟수
- market_segment_Online TA : 시장 세분화 유형 중 "온라인 여행사"

위 변수들이 가장 중요도가 높은 것으로 나타났다.

(lead_time, adr, deposit_type_Non_Refund, deposit_type_No_Deposit 등 순서로 중요도가 높음을 의미한다.)

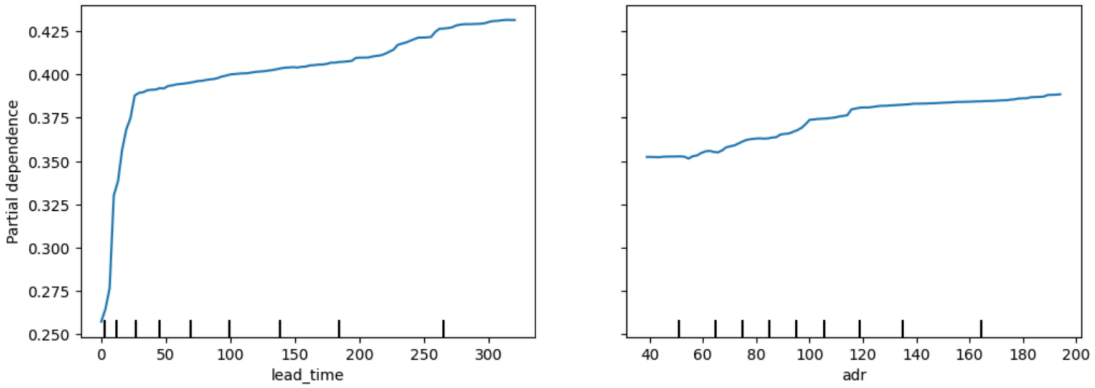


2.7.2 Partial Dependence Plots(PDP)

PDP는 모델에서 특정 특징(변수)들이 예측 결과에 어떤 영향을 미치는지 시각적으로 파악 가능하기 때문에 변수 중요도의 결과를 참고하여 중요도가 높은 순서로 수치형 변수에 대해서만 확인하였다.

아래의 lead_time 그림에서 **예약과 실제 이용 날짜 사이의 시간이 길수록** 사람들이 계획을 변경할 가능성이 더 높아지기 때문에 예측된 **취소 확률이 증가**하는 경향을 보이고 있음을 확인할 수 있다.

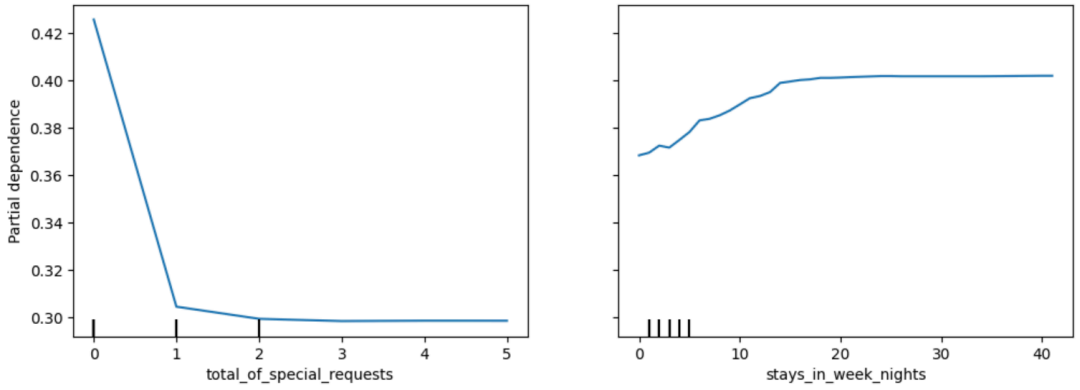
또한 adr 그림에서는 **하루 숙박 비용이 높을수록 예측된 취소 확률이 점차 증가**하는 경향을 보였다. 이는 고가의 예약이 경제적 부담이 될 수 있으며, 소비자가 더 나은 가격이나 조건을 찾아 예약을 변경하거나 취소할 가능성이 있는 것으로 해석할 수 있다.



<lead_time 과 adr의 PDP>

아래의 total_of_special_requests 그림에서는 **고객이 특별 요청을 많이 할수록 취소할 가능성이 낮아진다**는 것을 보여주며 이는 고객이 특정 서비스나 요구 사항에 대해 더 관심을 가지고 있어 그 예약을 유지할 가능성이 높다고 해석할 수 있다. 호텔이나 서비스 제공자는 이 정보를 사용하여 고객의 특별 요청을 적극적으로 수용함으로써 예약 취소율을 감소시킬 수 있을 것이다.

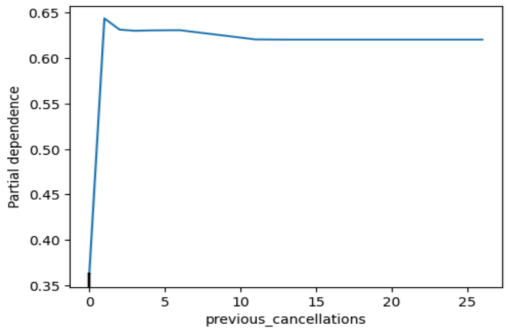
stays_in_week_nights 그림에서는 **더 긴 주중 숙박이 예약 취소와 관련이 있을 수 있으**며 이는 긴 숙박 기간 동안 발생할 수 있는 불확실성 또는 변경 가능성 때문일 수 있다고 해석할 수 있다. 이를 통해 호텔이나 숙박업체는 긴 숙박 예약에 대해 유연한 취소 정책을 제공하는 등의 정책을 세울 수 있을 것이다.



<total_of_special_requests 과 stays_in_week_nights의 PDP>

아래의 previous_cancellations 그림에서 **고객이 이전에 취소 이력이 있을 경우 다시 취소할 확률이 높아지고 초기 몇 회의 취소 후에는 추가 취소가 예측 확률에 미치는 영향이 크게 감소**한다. 이를 통해서 호텔이나 예약 서비스 제공자는 이전 취소 이력이 있는 고객을 관리하는 전략을 개발할 수 있다.

booking_changes 그림에서는 **예약 변경 횟수 0회 즉, 예약 변경이 없는 경우, 예약 취소 확률이 상대적으로 높다**. 이는 예약 초기에 불확실성이나 만족도가 낮을 경우 고객이 취소할 가능성이 높음을 의미할 수 있다. 예약 변경 횟수 1회 이상은 예약을 한 번 이상 변경할 경우, 예약 취소 확률이 급격히 감소한다. 이는 고객이 예약 변경을 통해 자신의 요구사항을 더 잘 충족시키게 되어 취소할 필요가 적어지기 때문이라고 해석할 수 있다



<previous_cancellations과 booking_changes의 PDP>

3. 결론

3.1 가설 확인

1.2.2 모델링
① 중요 변수 : 고객의 hotel booking cancel에 가장 많이 영향을 미치는 변수는 'lead_time'(예약 시점과 도착 시점 사이의 일수)인 것을 확인할 수 있었다. *가설이 틀렸다고 할 수 있다.*
② 적합 모델 : Random Forest와 XGBoost 중 Random Forest가 채택되었다. *가설이 맞다고 할 수 있다.*

3.2 의사결정 방향 제시

고객 유형 시각화를 통하여 얻은 호텔 예약 취소 예측에 영향을 미치는 변수들은 'lead_time', 'adr', 'total_of_special_requests', 'stays_in_week_nights', 'previous_cancellations', 'stays_in_weekend_nights', 'booking_changes' 인 것으로 나타났다. 이러한 변수들을 토대로 **실제 호텔들이 어떠한 전략을 펼칠 수 있을지 방향성을 제시**해보았다.

① lead_time
: 예약한 날짜와 이용하는 날짜 사이의 시간이 길수록 고객이 호텔 예약을 취소할 확률이 증가함을 알 수 있다. 호텔을 이용하는 날짜와 고객이 예약한 날짜의 간격이 큰 경우, **특정 기간을 정하여 고객에게 알림 문자를 전송하여 예약한 이용일에 실제로 호텔을 이용할 것인지에 대한 여부를 확인하는 전략**을 세울 수 있다.

② adr
: 호텔의 숙박 비용이 높을수록 고객이 호텔 예약을 취소할 확률이 높아짐을 알 수 있는데, 호텔의 숙박 비용의 경우 호텔이 자체적으로 **시설 혹은 서비스의 수준이 비슷한 다른 호텔과 가격을 비교하면서 조정하거나 할인쿠폰을 고객에게 증정**하는 등 상대적으로 고객이 저렴하게 이용할 수 있도록 하는 전략을 세울 수 있다.

③ booking_changes
: 고객의 예약 변경 횟수가 0회이면 고객이 호텔 예약을 취소할 확률이 상대적으로 높고, 고객의 예약 변경 횟수가 1회 이상이면 고객이 호텔 예약을 취소할 확률이 급격히 감소한다. **예약 변경 횟수가 0회인 고객들에게 알림 문자를 전송하여 예약일에 대해 상기시켜주고 추가적으로 호텔에게 바라는 서비스나 궁금한 점에 대한 설문을 진행하는 등의 서비스를 제공**할 수 있다.

④ previous_cancellations
: 고객이 이전에 호텔 예약을 취소한 이력이 있을 경우 다시 취소할 확률이 높으나 초기 몇 회의 취소 이후에는 추가 취소가 예측 확률에 미치는 영향이 크게 감소한다. 호텔에서 **취소 이력이 있는 고객들을 따로 관리하고, 고객이 예약 취소를 원할 경우 어떤 이유로 호텔 예약을 취소하는지 등을 조사**해야 한다. 고객의 단순변심으로 인한 취소인지 혹은 날짜 변경 등을 위한 취소인지 등을 조사할 수 있다.

⑤ stays_in_week_nights, stays_in_weekend_nights
: 주중/주말에 긴 숙박을 하는 경우 고객이 호텔 예약 취소를 할 확률이 높아진다. 한 고객이 주중/주말에 긴 숙박을 예약하고 노쇼를 하는 경우, 호텔이 받는 피해가 크기 때문에 **긴 숙박 예약을 원하는 경우 짧은 기간 동안의 숙박보다 예약금을 더 비싸게 받거나 취소 시에 수수료를 내는 기간을 더 타이트하게 잡는 등의 전략**을 세울 수 있다.

3.3 개선 방안

one-hot encoding 한계점

2.3.2 범주형 변수 전처리 과정에서 ‘reservation_status_date’ 열에대한 year과 month만 남기고 one-hot encoding을 진행하였다.

호텔 취소 예측을 목표로 하고있는 우리의 프로젝트에서는 원핫인코딩된 각 연도와 월의 중요도가 아닌 **연도와 월 그 자체의 변수 중요도**를 알고싶었다. 때문에 **원핫인코딩된 열을 하나로 합쳐 중요도 추출**에 사용하도록 시도해봤지만 해결하지 못하였다.

가중치 또는 Shapley value(모든 가능한 피처 부분 집합에 대한 평균 이익 기여도 (average marginal contribution))를 고려하여 기여도를 공정하게 할당 후 중요도 추출을 진행했더라면 다른 결과가 나올수도 있을 것이라고 생각한다.