

24.12.02 (Hypothesis_testing)

- 60210021 박세민

<최종 결론>

1. 적절한 numerical 독립변수

lead_time

stays_in_weekend_nights

stays_in_week_nights

adr

total_of_special_requests

arrival_date_day_of_month

6개의 numerical 컬럼(독립변수)이 hotel 컬럼(종속변수)와 연관성이 있다고 볼 수 있다.

2. 적절한 nominal 독립변수

meal

market_segment

distribution_channel

is_repeated_guest

deposit_type

customer_type

arrival_date_month

reserved_room_type

8개의 nominal 컬럼(독립변수)이 hotel 컬럼(종속변수)와 연관성이 있다고 볼 수 있다.

< Hypothesis_testing 진행 계획 및 variable type분석 >

1. Hypothesis_testing 목표: 가설 검정을 통해 회귀분석할 feature 선정하기

종속변수가 범주형일 때 (logistic regression을 위한 feature 선정)

step1) 해당 피쳐들을 기반으로 어떤 종류의 호텔(hotel: city hotel/resort hotel) 선호할 지 예측

step2) city hotel/resort hotel을 기준으로 연속형 컬럼들과 two sample t-test(독립성/정규성/등분산성 확인 필요)/범주형 컬럼들과 카이제곱 test 수행

step3) 이상치를 대체한 것, 대체하지 않은 것의 t-test 검정 수행 결과를 비교하여 이상치를 대체할 것인지 안 대체할 것인지 결정하기

2. variable type 판단하기

데이터셋 컬럼 정보(총 32개)

hotel: 호텔의 종류(Resort Hotel / City Hotel) - Nominal

is_canceled: 호텔 예약 취소 여부(1: 취소함 / 2: 취소 안함) - Binary(Nominal)

lead_time: 호텔 예약 날짜와 호텔 도착 날짜 사이 경과 일수 - Numeric

arrival_date_year: 호텔에 도착한 년도 - Date(Ordinal)

arrival_date_month: 호텔에 도착한 월 - Date

arrival_date_week_number: 호텔에 도착한 주 - Date

arrival_date_day_of_month: 호텔에 도착한 일 - Date

stays_in_weekend_nights: 주말에 숙박한 일 수 - Numeric(범주형 변환 가능)

stays_in_week_nights: 주중에 숙박한 일 수 - Numeric(범주형 변환 가능)

adults: 호텔에 방문한 어른 수 - Numeric

children: 호텔에 방문한 청소년 수 - Numeric

babies: 호텔에 방문한 유아 수 - Numeric

meal: 예약된 식사 유형(Undefined / SC는 식사가 없는 경우) - Nominal

country: 호텔을 예약한 손님의 국가 - Nominal

market_segment: 마켓 구분(TA: Travel Agents / TO: Tour Operator) (아래 컬럼이랑 비슷한 느낌인거 같은데, 더 찾아보겠습니다) - Nominal

distribution_channel: 예약 채널(TA: Travel Agents / TO: Tour Operator) (생산자에서 소비자에게 이르기까지의 방식, 채널) - Nominal

is_repeated_guest: 호텔 재방문 손님 여부(1: 맞음 / 2: 아님) - Binary(Nominal)

previous_cancellations: 해당 손님의 과거 예약 취소 횟수 - Numeric

previous_bookings_not_canceled: 해당 손님의 과거 예약 취소하지 않은 횟수 - Numeric

reserved_room_type: 예약된 객실 유형(익명성을 위해 명칭 대신 코드로 표시 'C' 'A' 'D' 'E' 'G' 'F' 'H' 'L' 'P' 'B') - Nominal

assigned_room_type: 예약에 배정된 객실 유형('C' 'A' 'D' 'E' 'G' 'F' 'I' 'B' 'H' 'P' 'L' 'K') - Nominal

booking_changes: 예약 후 예약 변경 혹은 수정 횟수 - Numeric

deposit_type: 보증금 지불 여부('No Deposit', 'Refundable', 'Non Refund') - Nominal

agent: 호텔 예약 여행 에이전시 ID - ID

company: 호텔 예약 담당 회사 ID - ID

days_in_waiting_list: 예약이 완료될 때까지 대기 목록에 속한 기간 - Numeric

customer_type: 고객 유형('Transient', 'Contract', 'Transient-Party', 'Group') - Nominal

adr: 평균 일일 숙박비용 - Numeric

required_car_parking_spaces: 고객이 요구한 주차공간 수 - Numeric

total_of_special_requests: 고객이 요청한 특별 요청 수(예: 트윈 침대, 높은 층수) - Numeric

reservation_status: 마지막 예약 상태('Check-Out', 'Canceled', 'No-Show') - Nominal

reservation_status_date: 마지막 예약 상태가 설정된 날짜 - date

3. 독립변수를 선택하는 기준

1. 도메인 지식 기반 선택

: 문제의 배경과 목표: 분석의 목적과 관련이 깊은 변수를 우선적으로 선택합니다.

예: 학생 성적 예측 모델에서는 공부 시간, 출석률 등이 중요할 가능성이 높습니다.

2. 독립변수와 종속변수 간의 연관성 분석(상관계수나 t-test, anova분석, 카이제곱 분석)

3. 독립변수 간 연관관계가 높은 변수 제거

: 독립변수 간 상관관계가 높은 경우, 대표 변수를 선택

4. (regression 파트=>) 회귀분석에 적합한 변수 스케일링, 로그 변환

< 종속변수가 범주형일 때 logistic regression을 위한 feature 선정 >

: logistic regression 가설: 해당 피쳐들을 기반으로 어떤 호텔(city hotel/resort hotel) 선호하는 지 예측 -> city hotel/resort hotel을 기준으로 연속형 컬럼들과 two sample t-test 수행(독립성/정규성/등분산성 확인 필요)

1. 도메인 지식 기반 선택

1) 적절한 독립변수 후보

-lead_time

예약과 도착 사이의 일수는 호텔 유형(리조트 vs. 시티)에 영향을 미칠 가능성이 큼.

예: 리조트는 보통 멀리 여행하거나 휴가 계획으로 방문하는 경우가 많아 예약 기간이 더 길 수 있음.

-stays_in_weekend_nights

주말 숙박일 수는 호텔 유형에 따라 차이가 있을 수 있습니다.

예: 리조트는 주말 숙박이 더 길어질 가능성이 높음.

-stays_in_week_nights

주중 숙박일 수는 시티 호텔에서 더 많을 가능성이 큼(출장이 많을 경우).

-adults, children, babies

방문 인원 구성(어른, 어린이, 유아)은 호텔 선택에 중요한 영향을 미칠 수 있습니다.

예: 가족 단위는 리조트를 선호하고, 비즈니스 여행객은 시티 호텔을 선호할 가능성.

-meal

예약된 식사 유형은 호텔 유형에 따라 다를 수 있습니다.

예: 리조트는 포함 식사 패키지가 더 일반적일 수 있음.

-market_segment

마켓 세그먼트는 고객의 예약 방식(TA, TO 등)과 관련이 있어 호텔 유형과 연관될 가능성이 있습니다.

-distribution_channel

예약 채널은 특정 호텔 유형과 강하게 연관될 수 있습니다.

-is_repeated_guest

재방문 여부는 고객의 충성도와 관련되며, 특정 호텔 유형에 대한 선호를 나타낼 수 있습니다.

-previous_cancellations, previous_bookings_not_canceled

과거 예약 기록은 고객의 호텔 선택 습관과 관련이 있을 수 있습니다.

-booking_changes

예약 변경 횟수는 호텔 유형에 따라 다를 수 있음(리조트는 계획 변경 가능성이 더 높을 수 있음).

-deposit_type

보증금 유형은 호텔의 비즈니스 모델(리조트 vs. 시티)과 관련이 있을 가능성이 있습니다.

-days_in_waiting_list

대기 시간은 호텔의 인기와 예약 구조를 반영하므로 호텔 유형과 관련될 수 있습니다.

-customer_type

고객 유형(Transient, Group 등)은 특정 호텔 유형과 연관될 가능성이 높습니다.

-adr (평균 일일 숙박비용)

숙박비용은 호텔의 특성과 밀접히 관련되어 있습니다. 일반적으로 리조트는 더 높은 일일 숙박비용을 가질 수 있음.

-required_car_parking_spaces

주차 공간 요구는 호텔 위치와 유형(리조트 vs. 시티)과 관련될 수 있습니다.

-total_of_special_requests

특별 요청 수는 고객의 목적과 선호를 나타내며, 호텔 선택에 영향을 줄 수 있습니다.

-arrival_date_year, month, week_number, day_of_month: 특정 날짜 정보는 호텔 선택에 영향을 줄 수 있습니다.

-reserved_room_type, assigned_room_type: 객실 유형은 호텔 선택에 영향을 줄 수 있습니다.

2) 제외하는 변수

다음 변수는 타겟 변수(hotel) 예측에 직접적으로 연관되지 않을 가능성이 크거나, 분석 목적과 관련이 낮아 제외할 수 있습니다.

-hotel: 타겟 변수.

-is_canceled, reservation_status, reservation_status_date: 호텔 선택 이후의 상태로, 호텔 선택 결정에는 영향을 주지 않음.

-agent, company: 예약 ID는 호텔 선택과 직접적인 연관성이 없거나 데이터 해석이 어렵습니다.

<t-test/chisquare test>

1. 독립변수와 종속변수 간의 연관성 분석

1. 아래의 표처럼 Numeric 타입은 t-test를 Nominal 타입은 카이제곱 검정을 진행할 예정이다.

변수명	변수 타입	적합한 분석 방법	설명
lead_time	Numeric	t-test, ANOVA	호텔 유형별 평균 예약 기간 차이를 비교 가능.
stays_in_weekend_nights	Numeric	t-test, ANOVA	주말 숙박일 수가 호텔 유형에 따라 차이가 있는지 분석 가능.
stays_in_week_nights	Numeric	t-test, ANOVA	주중 숙박일 수가 호텔 유형에 따라 차이가 있는지 분석 가능.
adults	Numeric	t-test, ANOVA	어른 방문자 수의 평균이 호텔 유형별로 다른지 확인 가능.
children	Numeric	t-test, ANOVA	어린이 방문자 수의 평균이 호텔 유형별로 다른지 확인 가능.
babies	Numeric	t-test, ANOVA	유아 방문자 수의 평균이 호텔 유형별로 다른지 확인 가능.
meal	Nominal	카이제곱 검정	예약된 식사 유형이 호텔 유형과 연관이 있는지 확인 가능.
market_segment	Nominal	카이제곱 검정	시장 세그먼트가 호텔 유형과 연관이 있는지 확인 가능.
distribution_channel	Nominal	카이제곱 검정	예약 채널이 호텔 유형과 연관이 있는지 확인 가능.

is_repeated_guest	Binary (Nominal)	카이제곱 검 정, t-test	재방문 여부가 호텔 유형과 연관이 있는지 또는 평균 차이가 있는지 확 인 가능.
previous_cancellations	Numeric	t-test, ANOVA	과거 예약 취소 횟수의 평균이 호텔 유형별로 다른지 확인 가능.
previous_bookings_not_canceled	Numeric	t-test, ANOVA	과거 비취소 예약 횟수의 평균이 호 텔 유형별로 다른지 확인 가능.
booking_changes	Numeric	t-test, ANOVA	예약 변경 횟수가 호텔 유형별로 다 른지 확인 가능.
deposit_type	Nominal	카이제곱 검 정	보증금 유형이 호텔 유형과 연관이 있는지 확인 가능.
days_in_waiting_list	Numeric	t-test, ANOVA	대기 목록 일수의 평균이 호텔 유형 별로 다른지 확인 가능.
customer_type	Nominal	카이제곱 검 정	고객 유형이 호텔 유형과 연관이 있 는지 확인 가능.
adr	Numeric	t-test, ANOVA	평균 숙박비용이 호텔 유형별로 다른 지 확인 가능.
required_car_parking_spaces	Numeric	t-test, ANOVA	주차 공간 요구가 호텔 유형별로 다 른지 확인 가능.
total_of_special_requests	Numeric	t-test, ANOVA	특별 요청 수의 평균이 호텔 유형별 로 다른지 확인 가능.



2. 이상치를 중위수로 대체한 two sample t-test function 구현

1) Numeric Variables의 이상치 대체

1) 확인 방법: IQR을 사용하여 이상치를 탐지.(EDA&Preprocessing 파일 참고)

2) 처리 선택:

-삭제: 종속변수와 연관이 적거나, 데이터가 매우 왜곡된 경우.

-중위수 대체: 데이터 분포가 비대칭인 경우.

-평균 대체: 데이터가 정규분포에 가까운 경우.

```
from scipy.stats import kstest, bartlett, ttest_ind, ranksums

print("=== T-test 및 정규성/등분산성 검토 ===")
for var in numeric_cols:
    if df[var].dtype in ['int64', 'float64']:
        # 두 그룹 생성
        group1 = df[df['hotel'] == 'Resort Hotel'][var].copy()
        group2 = df[df['hotel'] == 'City Hotel'][var].copy()

        # IQR 계산 및 이상치 대체
        for group, name in zip([group1, group2], ['Group1', 'Group2']):
            Q1 = group.quantile(0.25)
            Q3 = group.quantile(0.75)
            IQR = Q3 - Q1
            lower_bound = Q1 - 1.5 * IQR
            upper_bound = Q3 + 1.5 * IQR

            # median 계산 (IQR 내부 값만 사용)
            median_value = group[(group >= lower_bound) & (group <= upper_bound)].median()

            # 이상치를 평균값으로 대체
            group[(group < lower_bound) | (group > upper_bound)] = median_value
            print(f"{var} - {name} 이상치 처리 완료. 중앙값 = {median_value:.4f}")

        # 정규성 검정 (shapiro test -> KS test로 대체)
        stat_g1, p_g1 = kstest(group1, 'norm', args=(group1.mean(), group1.std()))
        stat_g2, p_g2 = kstest(group2, 'norm', args=(group2.mean(), group2.std()))
        is_normal = (p_g1 > 0.05) and (p_g2 > 0.05)
        print(f"{var} - 정규성 검정: Group1 p-value = {p_g1:.10f}, Group2 p-value = {p_g2:.10f}, Normal = {is_normal}")

        # 등분산성 검정 (bartlett test)
        stat_var, p_var = bartlett(group1, group2)
        is_equal_var = p_var > 0.05
        print(f"{var} - 등분산성 검정: Bartlett p-value = {p_var:.10f}, Equal Variance = {is_equal_var}")

        # T-test 또는 대안적 방법 선택
        if is_normal:
            if is_equal_var: # 등분산 two sample t-test
                t_stat, p_value = ttest_ind(group1, group2, equal_var=True)
                test_type = "T-test (Equal Variance)"
            else: # 이분산 two sample t-test
                t_stat, p_value = ttest_ind(group1, group2, equal_var=False)
                test_type = "T-test (Unequal Variance)"
        else: # 비모수 검정
            t_stat, p_value = ranksums(group1, group2)
            test_type = "Rank sums Test (Non-Normal)"

        print(f"{var}: {test_type}, t-statistic = {t_stat:.4f}, p-value = {p_value:.10f}")
print("-" * 50)
```


3. 이상치를 대체하지 않은 two sample t-test function 구현

```
[23]: from scipy.stats import kstest, bartlett, ttest_ind, ranksums

print("=== T-test 및 정규성/등분산성 검토 ===")
for var in numeric_cols:
    if df[var].dtype in ['int64', 'float64']:
        group1 = df[df['hotel'] == 'Resort Hotel'][var].dropna()
        group2 = df[df['hotel'] == 'City Hotel'][var].dropna()
        print(np.shape(group1), np.shape(group2))

        # 정규성 검정 (shapiro test-> KS test로 대체)
        stat_g1, p_g1 = kstest(group1, 'norm', args=(group1.mean(), group1.std()))
        stat_g2, p_g2 = kstest(group2, 'norm', args=(group2.mean(), group2.std()))
        is_normal = (p_g1 > 0.05) and (p_g2 > 0.05)
        print(f"{var} - 정규성 검정: Group1 p-value = {p_g1:.10f}, Group2 p-value = {p_g2:.10f}, Normal = {is_normal}")

        # 등분산성 검정 (bartlett test)
        stat_var, p_var = bartlett(group1, group2)
        is_equal_var = p_var > 0.05
        print(f"{var} - 등분산성 검정: Bartlett p-value = {p_var:.10f}, Equal Variance = {is_equal_var}")

        # T-test 또는 대안적 방법 선택
        if is_normal:
            if is_equal_var: # 등분산 two sample t-test
                t_stat, p_value = ttest_ind(group1, group2, equal_var=True)
                test_type = "T-test (Equal Variance)"
            else: # 이분산 two sample t-test
                t_stat, p_value = ttest_ind(group1, group2, equal_var=False)
                test_type = "T-test (Unequal Variance)"
        else: # 비모수 검정
            t_stat, p_value = ranksums(group1, group2)
            test_type = "Rank sums Test (Non-Normal)"

        print(f"{var}: {test_type}, t-statistic = {t_stat:.4f}, p-value = {p_value:.10f}")
print("-" * 50)
```

*참고: t-test 중 shapiro를 사용하지 않은 이유

Shapiro-Wilk 테스트는 정규성 검정을 수행하기 위한 방법이지만, 표본 크기가 5000을 초과하면 p-value 계산의 정확도가 떨어질 수 있다고 경고를 발생Shapiro-Wilk 테스트는 정규성 검정을 수행하기 위한 방법이지만, 표본 크기가 5000을 초과하면 p-value 계산의 정확도가 떨어질 수 있다고 경고를 발생 -> KS 테스트 사용

*참고: t-test 중 levene을 사용하지 않은 이유

정규성을 안 따르면 등분산성의 여부와 상관없이 바로 비모수검정을 수행할 수 있다. 따라서 정규성이 검증 가능할 때만을 염두하고, bartlett을 사용했다.

3. two sample t-test function 수행 결과 비교 및 분석

(이상치 대체한 t-test 결과)

```
=== T-test 및 정규성/등분산성 검토 ===
lead_time - Group1 이상치 처리 완료. 중앙값 = 56.0000
lead_time - Group2 이상치 처리 완료. 중앙값 = 70.0000
lead_time - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
lead_time - 등분산성 검정: Bartlett p-value = 0.8458593703, Equal Variance = True
lead_time: Rank sums Test (Non-Normal), t-statistic = -24.5550, p-value = 0.0000000000
-----

stays_in_weekend_nights - Group1 이상치 처리 완료. 중앙값 = 1.0000
stays_in_weekend_nights - Group2 이상치 처리 완료. 중앙값 = 1.0000
stays_in_weekend_nights - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
stays_in_weekend_nights - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
stays_in_weekend_nights: Rank sums Test (Non-Normal), t-statistic = 53.1165, p-value = 0.0000000000
-----

stays_in_week_nights - Group1 이상치 처리 완료. 중앙값 = 3.0000
stays_in_week_nights - Group2 이상치 처리 완료. 중앙값 = 2.0000
stays_in_week_nights - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
stays_in_week_nights - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
stays_in_week_nights: Rank sums Test (Non-Normal), t-statistic = 63.3528, p-value = 0.0000000000
-----

adults - Group1 이상치 처리 완료. 중앙값 = 2.0000
adults - Group2 이상치 처리 완료. 중앙값 = 2.0000
adults - 정규성 검정: Group1 p-value = nan, Group2 p-value = nan, Normal = False
adults - 등분산성 검정: Bartlett p-value = nan, Equal Variance = False
adults: Rank sums Test (Non-Normal), t-statistic = 0.0000, p-value = 1.0000000000
-----

children - Group1 이상치 처리 완료. 중앙값 = 0.0000
children - Group2 이상치 처리 완료. 중앙값 = 0.0000
children - 정규성 검정: Group1 p-value = nan, Group2 p-value = nan, Normal = False
children - 등분산성 검정: Bartlett p-value = nan, Equal Variance = False
children: Rank sums Test (Non-Normal), t-statistic = 0.0000, p-value = 1.0000000000
-----

babies - Group1 이상치 처리 완료. 중앙값 = 0.0000
babies - Group2 이상치 처리 완료. 중앙값 = 0.0000
babies - 정규성 검정: Group1 p-value = nan, Group2 p-value = nan, Normal = False
babies - 등분산성 검정: Bartlett p-value = nan, Equal Variance = False
babies: Rank sums Test (Non-Normal), t-statistic = 0.0000, p-value = 1.0000000000
-----

previous_cancellations - Group1 이상치 처리 완료. 중앙값 = 0.0000
previous_cancellations - Group2 이상치 처리 완료. 중앙값 = 0.0000
previous_cancellations - 정규성 검정: Group1 p-value = nan, Group2 p-value = nan, Normal = False
previous_cancellations - 등분산성 검정: Bartlett p-value = nan, Equal Variance = False
previous_cancellations: Rank sums Test (Non-Normal), t-statistic = 0.0000, p-value = 1.0000000000
-----

previous_bookings_not_canceled - Group1 이상치 처리 완료. 중앙값 = 0.0000
previous_bookings_not_canceled - Group2 이상치 처리 완료. 중앙값 = 0.0000
previous_bookings_not_canceled - 정규성 검정: Group1 p-value = nan, Group2 p-value = nan, Normal = False
previous_bookings_not_canceled - 등분산성 검정: Bartlett p-value = nan, Equal Variance = False
previous_bookings_not_canceled: Rank sums Test (Non-Normal), t-statistic = 0.0000, p-value = 1.0000000000
-----
```

```

-----
booking_changes - Group1 이상치 처리 완료. 중앙값 = 0.0000
booking_changes - Group2 이상치 처리 완료. 중앙값 = 0.0000
booking_changes - 정규성 검정: Group1 p-value = nan, Group2 p-value = nan, Normal = False
booking_changes - 등분산성 검정: Bartlett p-value = nan, Equal Variance = False
booking_changes: Rank sums Test (Non-Normal), t-statistic = 0.0000, p-value = 1.0000000000
-----
days_in_waiting_list - Group1 이상치 처리 완료. 중앙값 = 0.0000
days_in_waiting_list - Group2 이상치 처리 완료. 중앙값 = 0.0000
days_in_waiting_list - 정규성 검정: Group1 p-value = nan, Group2 p-value = nan, Normal = False
days_in_waiting_list - 등분산성 검정: Bartlett p-value = nan, Equal Variance = False
days_in_waiting_list: Rank sums Test (Non-Normal), t-statistic = 0.0000, p-value = 1.0000000000
-----
adr - Group1 이상치 처리 완료. 중앙값 = 73.7950
adr - Group2 이상치 처리 완료. 중앙값 = 99.0000
adr - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
adr - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
adr: Rank sums Test (Non-Normal), t-statistic = -91.1762, p-value = 0.0000000000
-----
required_car_parking_spaces - Group1 이상치 처리 완료. 중앙값 = 0.0000
required_car_parking_spaces - Group2 이상치 처리 완료. 중앙값 = 0.0000
required_car_parking_spaces - 정규성 검정: Group1 p-value = nan, Group2 p-value = nan, Normal = False
required_car_parking_spaces - 등분산성 검정: Bartlett p-value = nan, Equal Variance = False
required_car_parking_spaces: Rank sums Test (Non-Normal), t-statistic = 0.0000, p-value = 1.0000000000
-----
total_of_special_requests - Group1 이상치 처리 완료. 중앙값 = 0.0000
total_of_special_requests - Group2 이상치 처리 완료. 중앙값 = 0.0000
total_of_special_requests - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
total_of_special_requests - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
total_of_special_requests: Rank sums Test (Non-Normal), t-statistic = 12.6722, p-value = 0.0000000000
-----
arrival_date_week_number - Group1 이상치 처리 완료. 중앙값 = 28.0000
arrival_date_week_number - Group2 이상치 처리 완료. 중앙값 = 27.0000
arrival_date_week_number - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
arrival_date_week_number - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
arrival_date_week_number: Rank sums Test (Non-Normal), t-statistic = -0.4367, p-value = 0.6623500042
-----
arrival_date_day_of_month - Group1 이상치 처리 완료. 중앙값 = 16.0000
arrival_date_day_of_month - Group2 이상치 처리 완료. 중앙값 = 16.0000
arrival_date_day_of_month - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
arrival_date_day_of_month - 등분산성 검정: Bartlett p-value = 0.0000458624, Equal Variance = False
arrival_date_day_of_month: Rank sums Test (Non-Normal), t-statistic = 0.5942, p-value = 0.5523584922

```

(이상치를 대체하지 않은 t-test 결과)

```
=== T-test 및 정규성/등분산성 검토 ===
(40060,) (79330,)
lead_time - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
lead_time - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
lead_time: Rank sums Test (Non-Normal), t-statistic = -29.4530, p-value = 0.0000000000
-----
(40060,) (79330,)
stays_in_weekend_nights - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
stays_in_weekend_nights - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
stays_in_weekend_nights: Rank sums Test (Non-Normal), t-statistic = 53.8082, p-value = 0.0000000000
-----
(40060,) (79330,)
stays_in_week_nights - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
stays_in_week_nights - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
stays_in_week_nights: Rank sums Test (Non-Normal), t-statistic = 61.1211, p-value = 0.0000000000
-----
(40060,) (79330,)
adults - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
adults - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
adults: Rank sums Test (Non-Normal), t-statistic = 1.8900, p-value = 0.0587543453
-----
(40060,) (79330,)
children - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
children - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
children: Rank sums Test (Non-Normal), t-statistic = 6.4847, p-value = 0.0000000001
-----
(40060,) (79330,)
babies - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
babies - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
babies: Rank sums Test (Non-Normal), t-statistic = 2.5511, p-value = 0.0107387255
-----
(40060,) (79330,)
previous_cancellations - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
previous_cancellations - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
previous_cancellations: Rank sums Test (Non-Normal), t-statistic = -11.3975, p-value = 0.0000000000
-----
(40060,) (79330,)
previous_bookings_not_canceled - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
previous_bookings_not_canceled - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
previous_bookings_not_canceled: Rank sums Test (Non-Normal), t-statistic = 8.6087, p-value = 0.0000000000
-----
(40060,) (79330,)
booking_changes - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
booking_changes - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
booking_changes: Rank sums Test (Non-Normal), t-statistic = 18.5747, p-value = 0.0000000000
-----
```

```

-----
(40060,) (79330,)
days_in_waiting_list - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
days_in_waiting_list - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
days_in_waiting_list: Rank sums Test (Non-Normal), t-statistic = -10.4441, p-value = 0.0000000000
-----
(40060,) (79330,)
adr - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
adr - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
adr: Rank sums Test (Non-Normal), t-statistic = -73.6647, p-value = 0.0000000000
-----
(40060,) (79330,)
required_car_parking_spaces - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
required_car_parking_spaces - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
required_car_parking_spaces: Rank sums Test (Non-Normal), t-statistic = 31.8684, p-value = 0.0000000000
-----
(40060,) (79330,)
total_of_special_requests - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
total_of_special_requests - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
total_of_special_requests: Rank sums Test (Non-Normal), t-statistic = 13.8582, p-value = 0.0000000000
-----
(40060,) (79330,)
arrival_date_week_number - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
arrival_date_week_number - 등분산성 검정: Bartlett p-value = 0.0000000000, Equal Variance = False
arrival_date_week_number: Rank sums Test (Non-Normal), t-statistic = -0.4367, p-value = 0.6623500042
-----
(40060,) (79330,)
arrival_date_day_of_month - 정규성 검정: Group1 p-value = 0.0000000000, Group2 p-value = 0.0000000000, Normal = False
arrival_date_day_of_month - 등분산성 검정: Bartlett p-value = 0.0000458624, Equal Variance = False
arrival_date_day_of_month: Rank sums Test (Non-Normal), t-statistic = 0.5942, p-value = 0.5523584922
-----

```

1) 이상치 대체한 t-test 결과와 대체하지 않은 t-test 결과 비교 분석

t-test 결과(t statistic/p value)가 크게 다를 경우 -> 이상치 대체가 필요함

: 이상치를 대체한 결과가 더 신뢰성이 높아 보인다면(예: p-value가 극단적이거나 더 합리적인 결과를 도출할 경우), 이상치 대체한 데이터를 사용하는 것이 좋다. 이상치는 분석 결과를 왜곡할 수 있기 때문에, 대체가 더 안정적인 결론을 제공할 수 있다.

t-test 결과(t statistic/p value)가 크게 다르지 않을 경우 -> 이상치 대체가 필요하지 않음

: 대체 없이도 안정적인 결과가 나온다면, 원 데이터를 사용하는 것이 더 자연스러울 수 있다. 이상치가 분석의 중요한 특성을 반영할 수도 있기 때문에, 대체 없이 진행하는 것이 데이터의 원래 의미를 보존할 수 있다.

결론

: children, babies, previous cancellations, previous bookings not canceled, booking changes, days in waiting list, required car parking spaces의 변수는 기각 결과가 달라짐->t-test결과가 크게 달라졌기 때문에 이상치를 대체한 결과를 사용해야함.

그 외의 변수들에서 p-value와 t-statistic 차이가 미미하기 때문에 원 데이터를 이용하는 결과를 사용해야함.

2) 적절한 numerical 독립변수

lead_time

stays_in_weekend_nights

stays_in_week_nights

adr

total_of_special_requests

arrival_date_day_of_month

6개의 numerical 컬럼(독립변수)이 hotel 컬럼(종속변수)와 연관성이 있다고 볼 수 있다.

4. chisquare function 만들기(결측치는 Preprocessing된 상태)

```
[18]: #nominal 독립변수와의 연관성 - 카이제곱 test
from scipy.stats import chi2_contingency
import pandas as pd

print("\n=== 카이제곱 검정 결과 ===")
for var in nominal_cols:
    # 결측값 제거
    data_clean = df.dropna(subset=[var, 'hotel'])

    # 교차표 생성
    contingency_table = pd.crosstab(data_clean[var], data_clean['hotel'])

    # 카이제곱 검정
    chi2, p, dof, expected = chi2_contingency(contingency_table)
    print(f"{var}: chi2 = {chi2:.4f}, p-value = {p:.4f}")
    print("Expected Frequencies:\n", expected)
```

5. chisquare function 수행하기

```
=== 카이제곱 검정 결과 ===
meal: chi2 = 11973.6428, p-value = 0.0000
Expected Frequencies:
[[61336.39584555 30973.60415445]
 [ 530.23988609  267.76011391]
 [ 9610.09958958  4852.90041042]
 [ 7076.50975794  3573.49024206]
 [ 776.75492085  392.24507915]]

-----
market_segment: chi2 = 2576.4052, p-value = 0.0000
Expected Frequencies:
[[1.57477259e+02 7.95227406e+01]
 [4.93694531e+02 2.49305469e+02]
 [3.51832105e+03 1.77667895e+03]
 [8.37619549e+03 4.22980451e+03]
 [1.31636371e+04 6.64736293e+03]
 [1.60925812e+04 8.12641880e+03]
 [3.75267645e+04 1.89502355e+04]
 [1.32892202e+00 6.71077980e-01]]

-----
distribution_channel: chi2 = 4177.8833, p-value = 0.0000
Expected Frequencies:
[[4.43660616e+03 2.24039384e+03]
 [9.73103149e+03 4.91396851e+03]
 [1.28240975e+02 6.47590250e+01]
 [6.50307991e+04 3.28392009e+04]
 [3.32230505e+00 1.67769495e+00]]

-----
is_repeated_guest: chi2 = 302.9122, p-value = 0.0000
Expected Frequencies:
[[76798.40355139 38781.59644861]
 [ 2531.59644861  1278.40355139]]

-----
deposit_type: chi2 = 3721.9807, p-value = 0.0000
Expected Frequencies:
[[6.95298646e+04 3.51111354e+04]
 [9.69249275e+03 4.89450725e+03]
 [1.07642684e+02 5.43573164e+01]]

-----
customer_type: chi2 = 320.9034, p-value = 0.0000
Expected Frequencies:
[[ 2708.34307731  1367.65692269]
 [ 383.39400285  193.60599715]
 [59544.34450121 30068.65549879]
 [16693.91841863 8430.08158137]]
```

```

-----
arrival_date_month: chi2 = 598.1306, p-value = 0.0000
Expected Frequencies:
[[7368.20814139 3720.79185861]
 [9220.72543764 4656.27456236]
 [4505.04564871 2274.95435129]
 [5360.87142977 2707.12657023]
 [3939.58932909 1989.41067091]
 [8412.74084932 4248.25915068]
 [7268.53898987 3670.46101013]
 [6507.73113326 3286.26886674]
 [7834.6597705 3956.3402295 ]
 [4514.34810286 2279.65189714]
 [7415.3848731 3744.6151269 ]
 [6982.1562945 3525.8437055 ]]
-----

reserved_room_type: chi2 = 12434.6485, p-value = 0.0000
Expected Frequencies:
[[5.71396601e+04 2.88543999e+04]
 [7.42867409e+02 3.75132591e+02]
 [6.19277661e+02 3.12722399e+02]
 [1.27583159e+04 6.44268414e+03]
 [4.34225270e+03 2.19274730e+03]
 [1.92494355e+03 9.72056454e+02]
 [1.39138136e+03 7.02618645e+02]
 [3.99341067e+02 2.01658933e+02]
 [3.98676606e+00 2.01323394e+00]
 [7.97353212e+00 4.02646788e+00]]
-----

```

적절한 nominal 독립변수

meal

market_segment

distribution_channel

is_repeated_guest

deposit_type

customer_type

arrival_date_month

reserved_room_type

8개의 nominal 컬럼(독립변수)이 hotel 컬럼(종속변수)와 연관성이 있다고 볼 수 있다.

