PRESENTATION

제주도 내국인 관광 목적 입도객 수 예측 프로그램

빅콘테스트

유지민
UUJIWINO598@NAVER.COM
(깃허브계정)

INDEX

01

데이터 불러오기

01

01 데이터 불러오기

본 대회에 사용되는 외부 데이터는 다음과 같다.

1. 기상청 2015년 01월 - 2022년 7월 데이터
2. 휴일 개수 데이터
3. 방학 데이터
4. 주말 개수 데이터
5. 계절 데이터

01 데이터 불러오기 [대표 컬럼 설명]

컬럼	의미
Vacation	방학
holiday	공휴일
season	계절
weekend	주말
Average temperature (°C)	평균기온(°C)
Average maximum temperature (°C)	평균최고기온(°C)
Average minimum temperature (°C)	평균최저기온(°C)
Maximum temperature (°C)	최고기온(°C)
Minimum temperature (°C)	최저기온(°C)
The day the highest temperature appeared (yyyymmdd)	최고기온 나타난날(yyyymmdd)

01 데이터 불러오기 [대표 컬럼 설명]

컬럼	의미
The day the lowest temperature appeared (yyyymmdd)	최저기온 나타난날(yyyymmdd)
Monthly precipitation (00~24hten thousand)(mm)	월합강수량(00~24h만)(mm))
Average wind speed (m/s)	평균풍속(m/s)
Maximum wind speed (m/s)	최대풍속(m/s)
The day the maximum wind speed appeared (yyyymmdd)	최대풍속 나타난날(yyyymmdd)
Average Cloudiness (1/10)	평균운량(1/10)
leis_sports_cnt	레저스포츠
ref_view_cnt	휴양 및 관람

01 데이터 불러오기 [코드]

```
df = pd.read_csv('빅콘.csv')
df
```

본 대회에 사용되는 데이터들은 구글 스프레드 시트로
데이터들을 미리 합친 결과이다.

02

EDA 분석

02

02 EDA 분석

본대회의 예측을 수행하기 위하여
중요하다고 생각하는 컬럼은
Vacation, holiday, season, weekend, Average temperature (°C),
Average maximum temperature (°C),
Average minimum temperature (°C),
Maximum temperature (°C),
Minimum temperature (°C),
Average wind speed (m/s)]과 같습니다

02 SEASON 컬럼 분석

기존의 형태는 모델을 학습시키기에
부적절한 형태이기 때문에
원핫인코딩을 통해 모델을 학습시키기 위한
전처리를 진행하고

각각의 계절과 목표로 하는
타겟 변수의 상관관계를 분석하고자 한다.

02 SEASON 컬럼 분석

season
winter
winter
spring
spring
spring
summer
summer
summer
autumn
autumn

season_spring	season_summer	season_autumn	season_winter
0	0	0	1
0	0	0	1
1	0	0	0
1	0	0	0
1	0	0	0
0	1	0	0
0	1	0	0
0	1	0	0
0	0	1	0
0	0	1	0

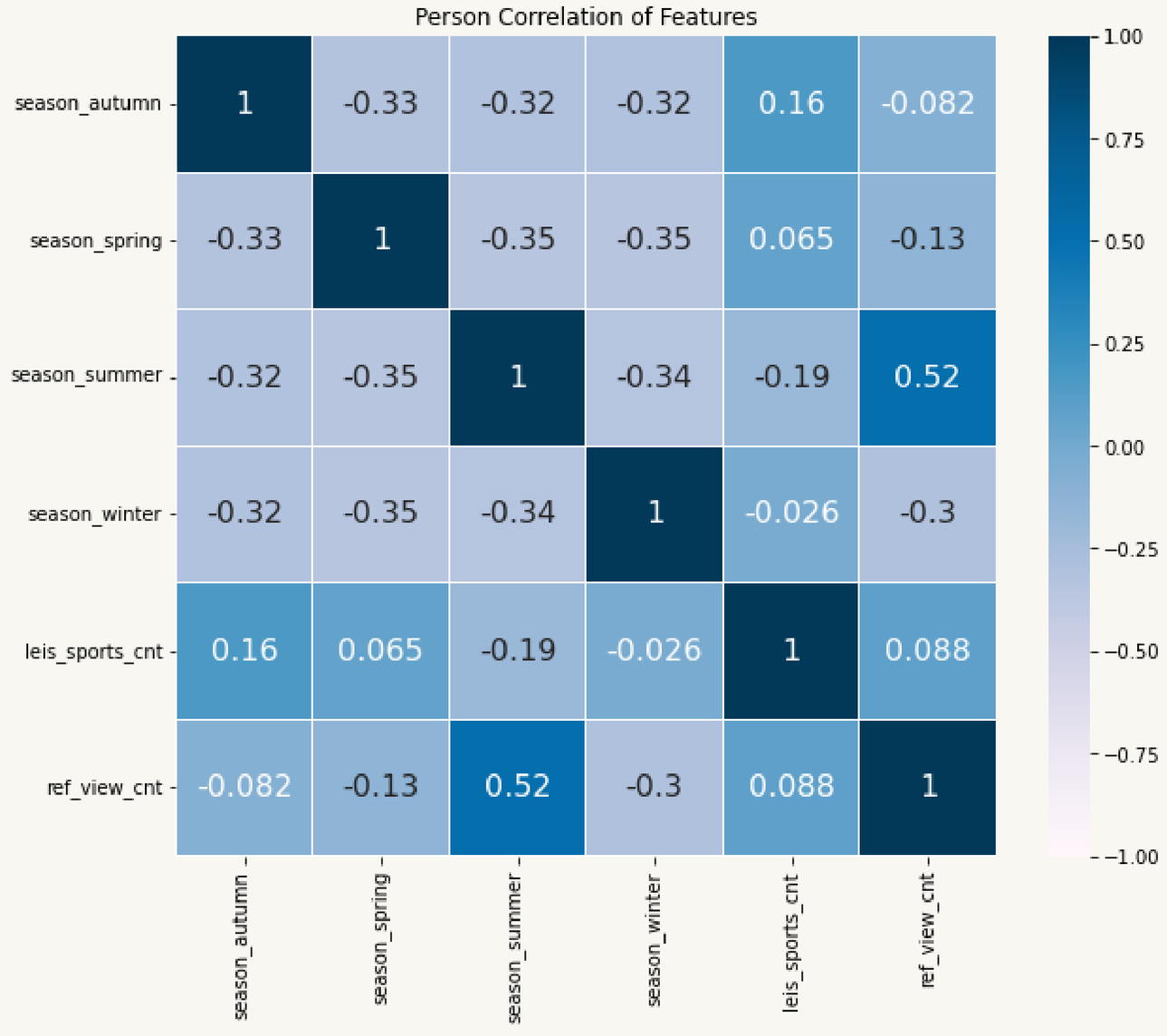
02 SEASON 컬럼 분석 [원핫이코딩 코드]

```
# season을 분석하기 위해서 원핫인코딩을 실행한다.

df = pd.get_dummies(df, columns = ['season'])

df
```

02 SEASON과 타겟 변수들 간에 상관관계 분석

	season_autumn	season_spring	season_summer	season_winter	leis_sports_cnt	ref_view_cnt
season_autumn	1.000000	-0.327815	-0.318544	-0.318544	0.159662	-0.082484
season_spring	-0.327815	1.000000	-0.348079	-0.348079	0.064811	-0.134630
season_summer	-0.318544	-0.348079	1.000000	-0.338235	-0.194176	0.518904
season_winter	-0.318544	-0.348079	-0.338235	1.000000	-0.026330	-0.302427
leis_sports_cnt	0.159662	0.064811	-0.194176	-0.026330	1.000000	0.087771
ref_view_cnt	-0.082484	-0.134630	0.518904	-0.302427	0.087771	1.000000

02 SEASON과 타겟 변수들 간에 상관관계 분석



02 SEASON 상관관계 분석 [코드]

상관관계 분석을 해보니
leis_sports_cnt(spring = 0.06, summer = -0.1, autumn = 0.1,
wintter = -0.02)하고는 관계가 있는 것이 없고,
# ref_view_cnt하고는
summer이 가장 관계가 깊고(0.5),
winter가 그 다음으로 반비례하여 높다.
(-0.3) 나머지는 상관관계가 없다.
(spring = -0.1, autumn = -0.08)

```
df_season = df[["season_autumn","season_spring","season_summer","seas
on_winter","leis_sports_cnt","ref_view_cnt"]]
df_season.corr()
```

02 VACATION 컬럼 분석

base_month	Vacation
201501	1
201502	0
201503	0
201504	0
201505	0
201506	1
201507	1
201508	1
201509	0
201510	0

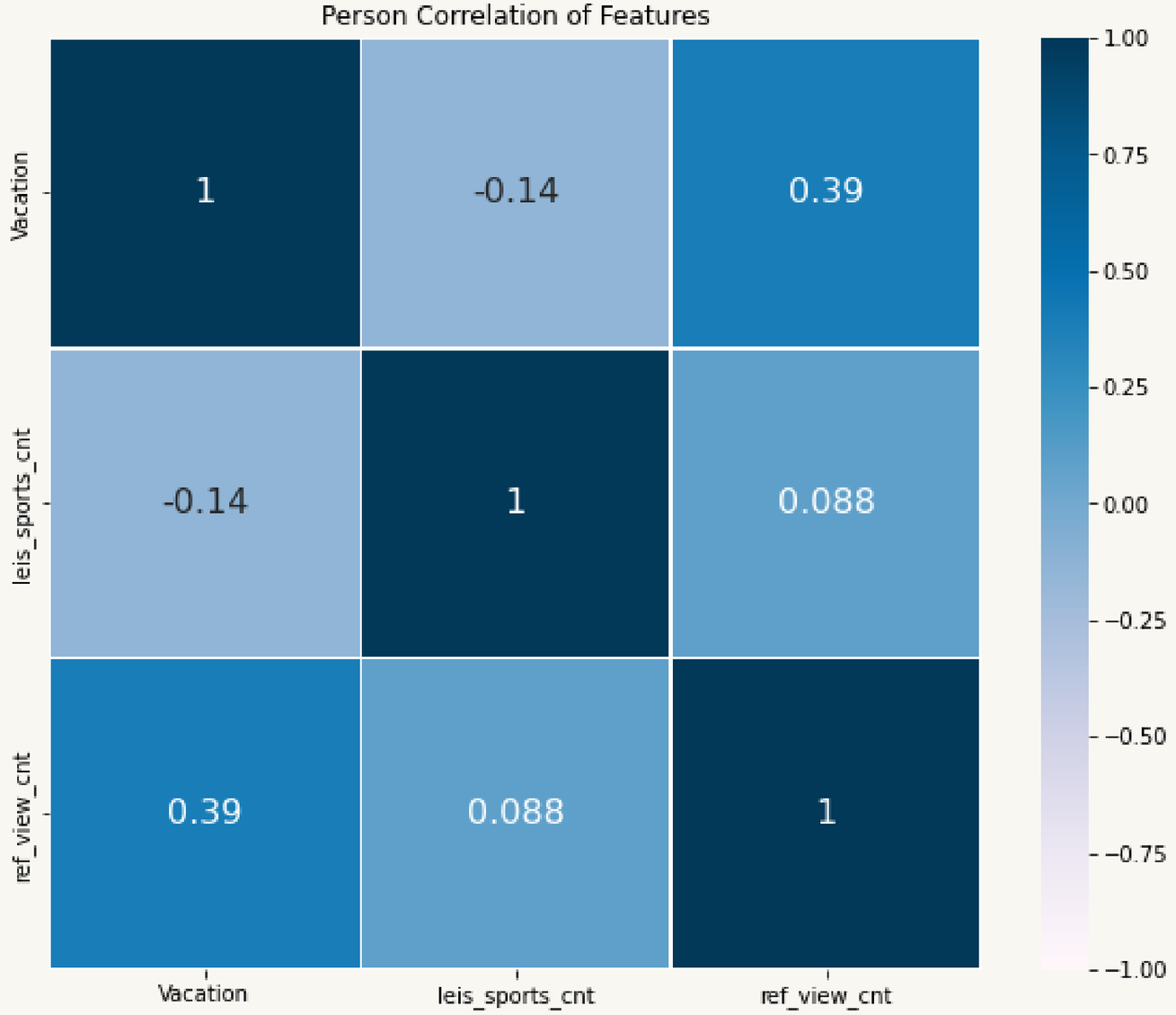
방학에는 제주도에
가족단위의 여행을
떠날 가능성이 높다고 생각하여
방학이 있는 달은 1로,
없는 달은 0으로
데이터를 매핑하였습니다.

02 VACATION 상관관계 분석 [코드]

상관관계 분석을 해 보니
vacation(0.01)과 ref_view_cnt(0.3)가 상관관계가 있고,
leis_sports_cnt(-0.1)는 상관관계가 없다.

```
df_vacation = df[["Vacation","leis_sports_cnt","ref_view_cnt"]]
df_vacation.corr()
```

02 VACATION과 타겟 변수들 간의 상관관계 분석



02 HOLIDAY 컬럼 분석

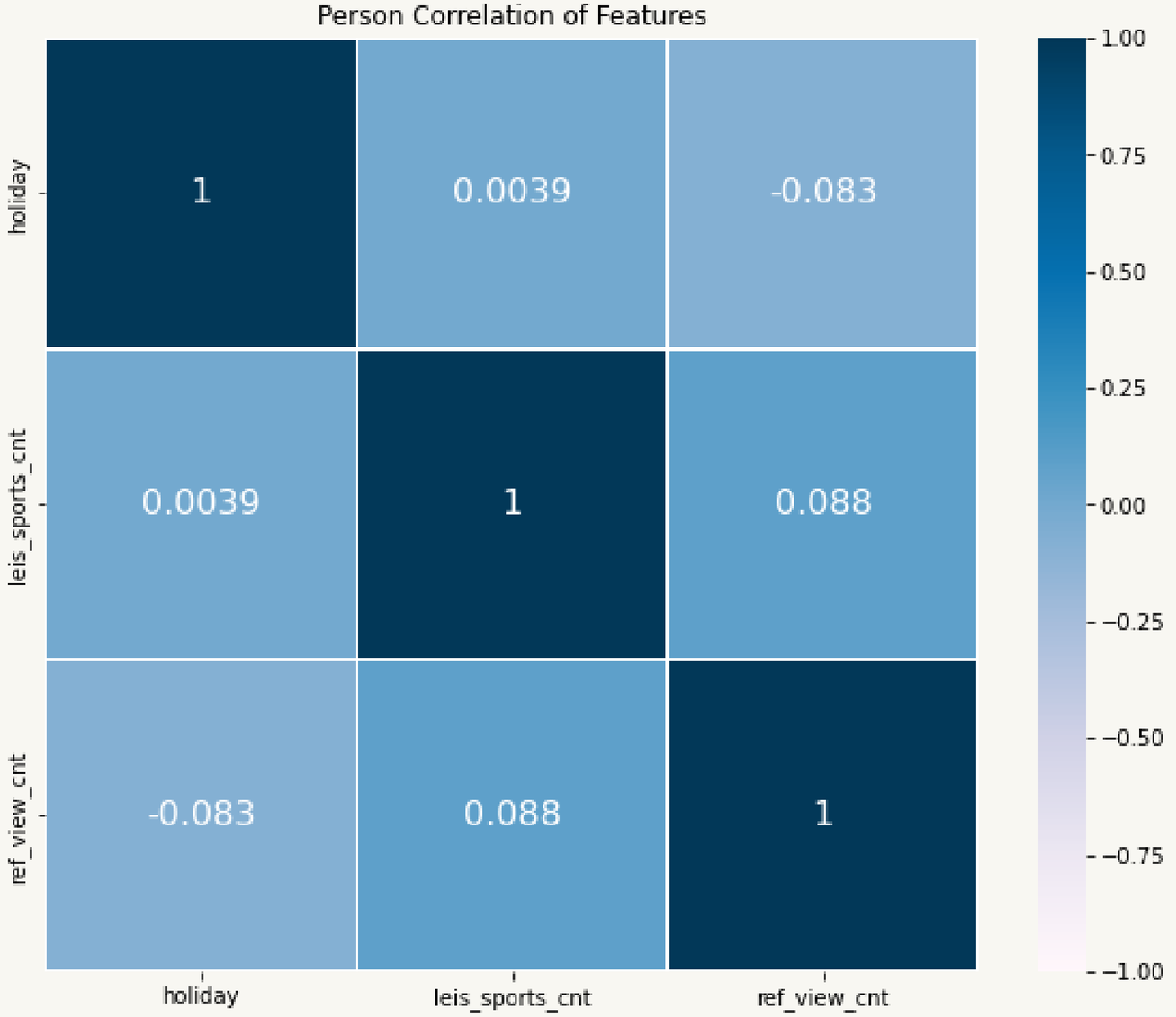
base_month	holiday
201501	1
201502	3
201503	1
201504	0
201505	2
201506	1
201507	0
201508	0
201509	4
201510	2

휴일에는 시간이 많아져
여행을 많이 떠날 거라
예상했습니다.

02 HOLIDAY와 타겟 변수들 간의 상관관계 분석

	holiday	leis_sports_cnt	ref_view_cnt
holiday	1.000000	0.003873	-0.083187
leis_sports_cnt	0.003873	1.000000	0.087771
ref_view_cnt	-0.083187	0.087771	1.000000

02 HOLIDAY와 타겟 변수들 간의 상관관계 분석



02 HOLIDAY 상관관계 분석 [코드]

상관관계 분석을 해 보니 holiday와 leis_sports_cnt(0.003),
ref_view_cnt(-0.08) 둘 다 관계가 없다.

```
df_holiday = df[["holiday","leis_sports_cnt","ref_view_cnt"]]
df_holiday.corr()
```

02 WEEKEND 컬럼 분석

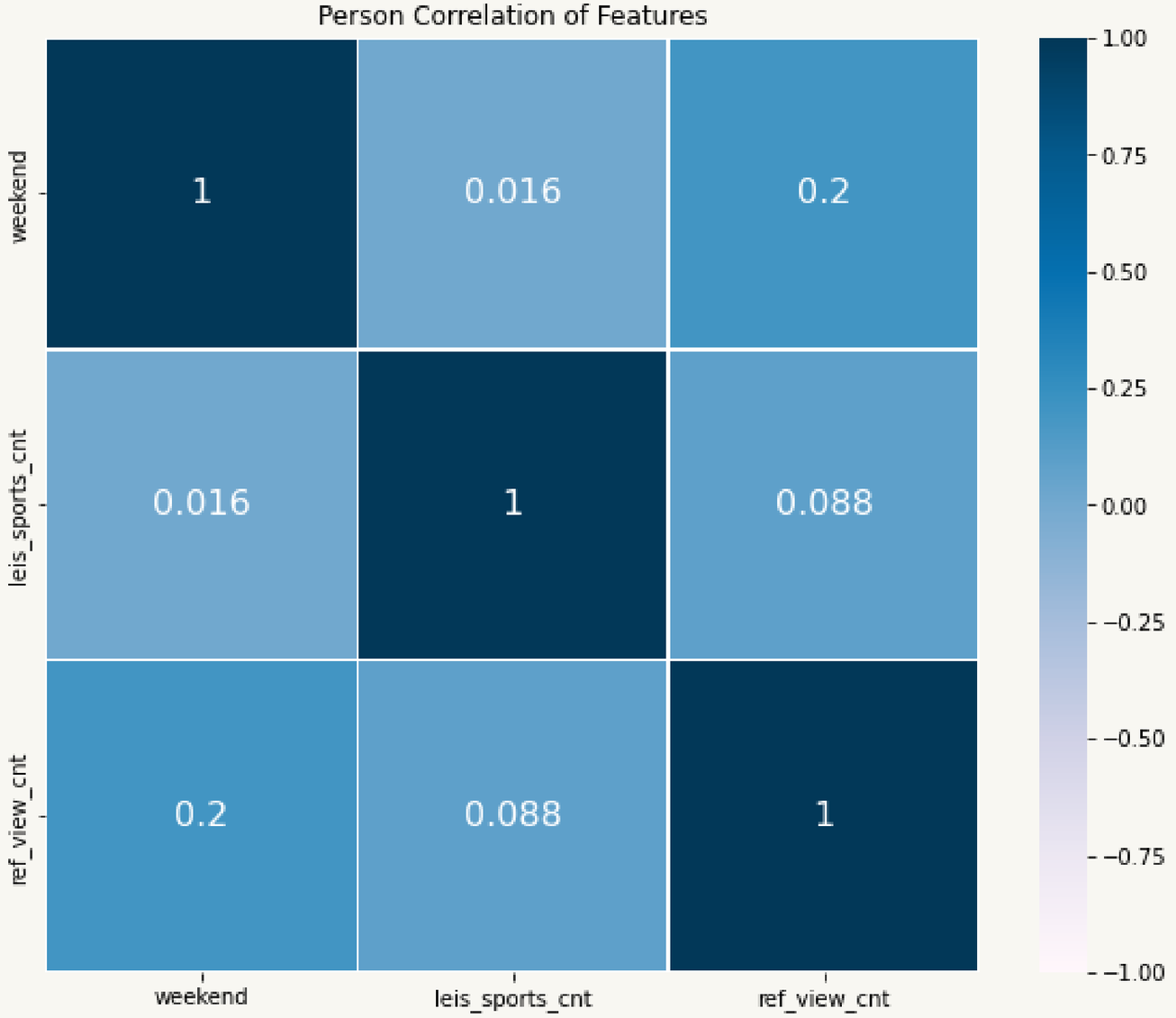
base_month	weekend
201501	9
201502	4
201503	8
201504	6
201505	10
201506	8
201507	8
201508	10
201509	7
201510	9

주말에는 시간이 많아
제주도로 많이
여행을 갈 것이라
예상했습니다.

02 WEEKEND와 타겟 변수들 간의 상관관계 분석

	weekend	leis_sports_cnt	ref_view_cnt
weekend	1.000000	0.015566	0.202462
leis_sports_cnt	0.015566	1.000000	0.087771
ref_view_cnt	0.202462	0.087771	1.000000

02 WEEKEND와 타겟 변수들 간의 상관관계 분석



02 WEEKEND 상관관계 분석 [코드]

상관관계 분석을 해 보니 weekend와 leis_sports_cnt(0.01),
ref_view_cnt(0.2) 둘 다 관계가 없다.

```
df_weekend = df[["weekend","leis_sports_cnt","ref_view_cnt"]]
df_weekend.corr()
```

02 AVERAGE TEMPERATURE (°C) 컬럼 분석

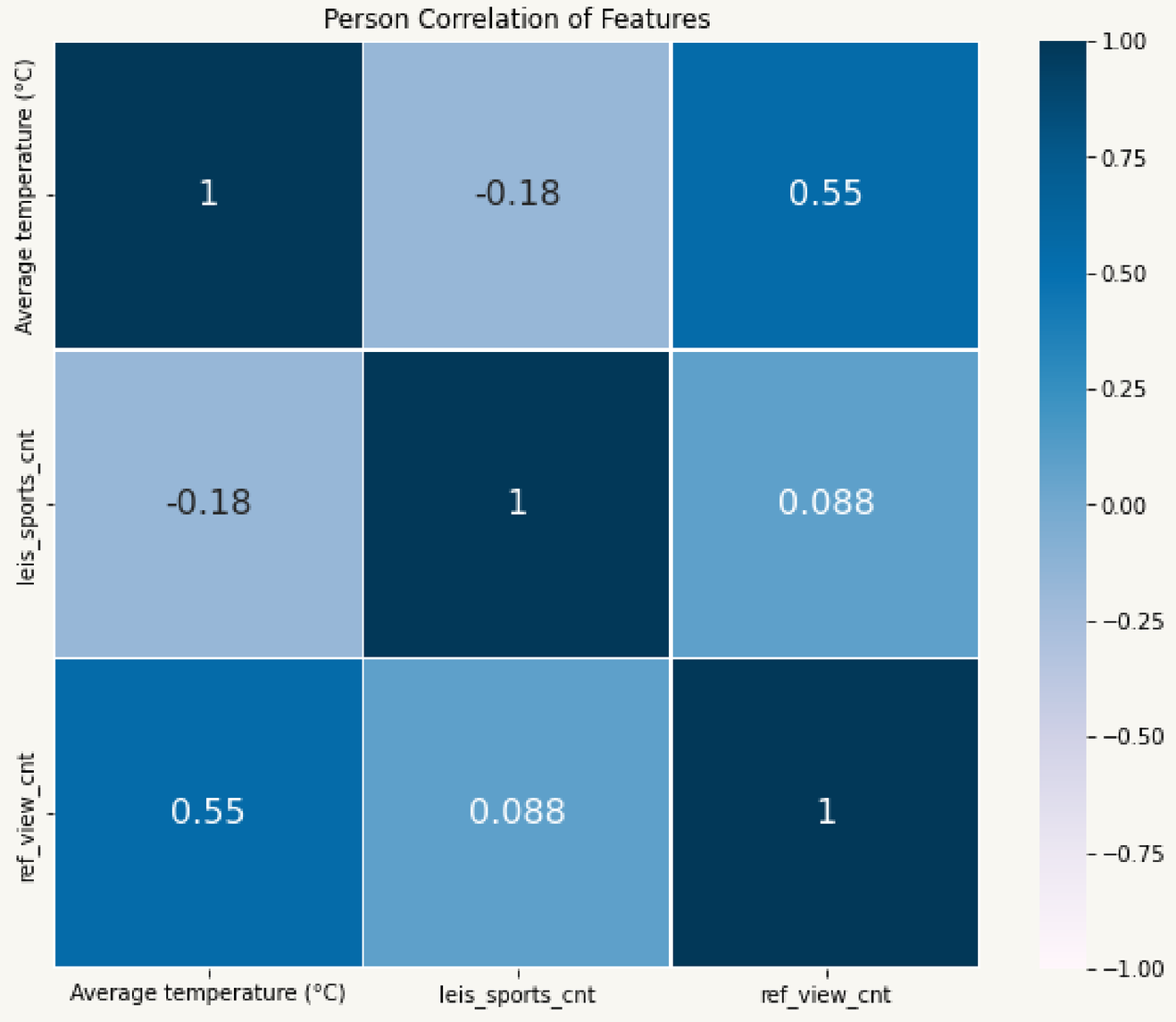
base_month	Average temperature (°C)
201501	7.4
201502	7.3
201503	10.4
201504	15.1
201505	18.8
201506	22
201507	25.6
201508	26.4
201509	23.2
201510	19.2

제주도는 섬이기 때문에
평균기온이 높으면
바다를 보러 놀러가는 사람들도
늘어날 것 이고,
레저스포츠를 즐기려는 사람도
늘어날 것 이라
생각했습니다.

02 AVERAGE TEMPERATURE (°C)와 타겟 변수들 간에 상관관계

index	Average temperature (°C)	leis_sports_cnt	ref_view_cnt
Average temperature (°C)	1.0	-0.17848774474875967	0.5521471379301875
leis_sports_cnt	-0.17848774474875967	1.0	0.08777131563384108
ref_view_cnt	0.5521471379301875	0.08777131563384108	1.0

02 AVERAGE TEMPERATURE (°C)와 타겟 변수들 간에 상관관계



02 AVERAGE TEMPERATURE (°C) 상관관계 분석 [코드]

상관관계 분석을 해 보니 Average temperature (°C)와
leis_sports_cnt(0.5),
ref_view_cnt(0.5) 둘 다 관계가 깊다.

```
df_Average_temperature = df[["Average temperature (°C)","leis_sports_
cnt","ref_view_cnt"]]
df_Average_temperature.corr()
```

02 AVERAGE MAXIMUM TEMPERATURE (℃) 컬럼 분석

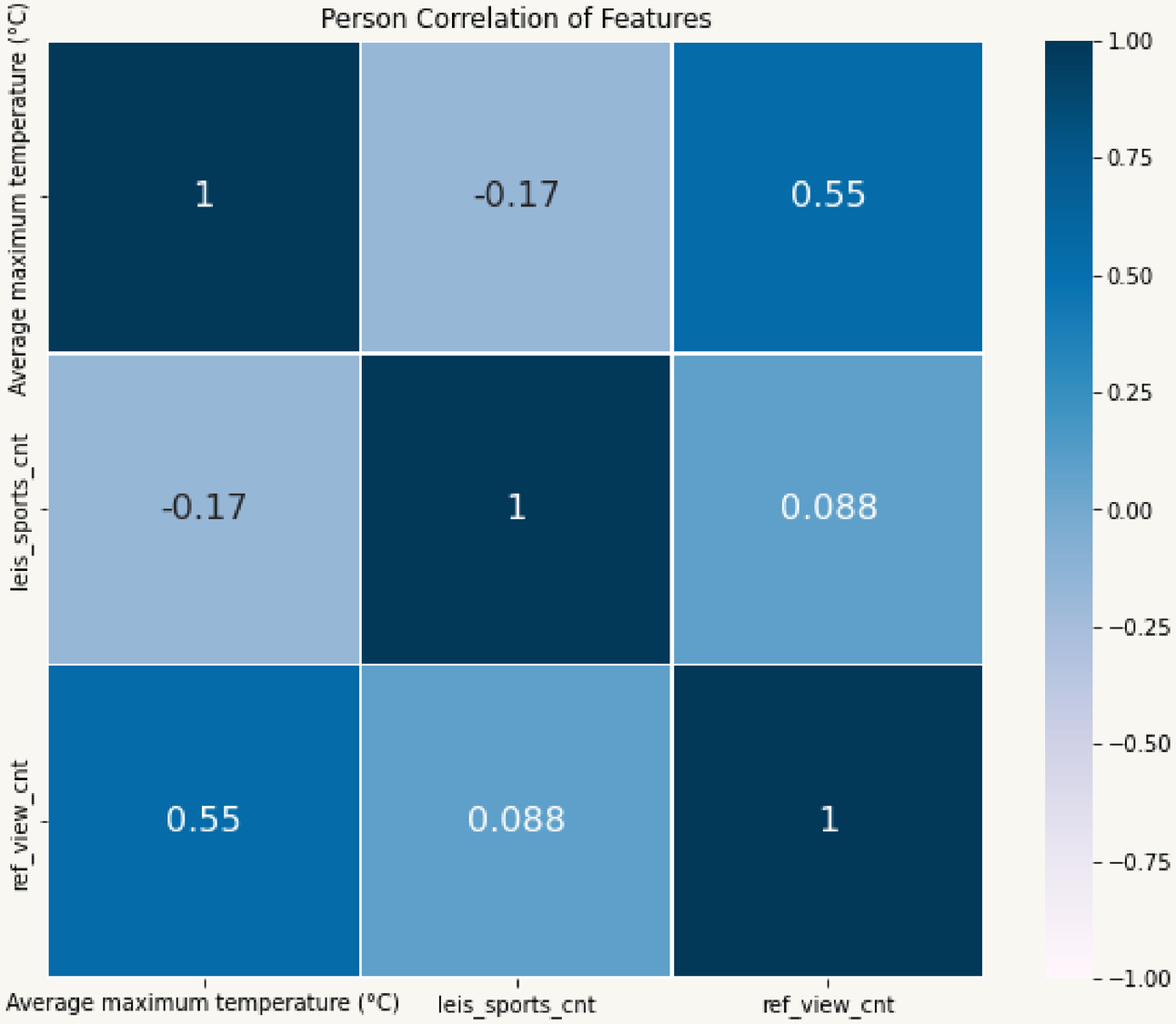
base_month	Average maximum temperature (°C)
201501	10.2
201502	10.1
201503	13.6
201504	19.2
201505	22.7
201506	25.3
201507	28.8
201508	29.3
201509	25.9
201510	22.6

기온이 높을 수록
바다를 보러 오는 여행객들이
늘것이라
예상했습니다.

02 AVERAGE MAXIMUM TEMPERATURE (°C)와 타겟 변수들 간의 상관관계

	Average maximum temperature (°C)	leis_sports_cnt	ref_view_cnt
Average maximum temperature (°C)	1.000000	-0.174064	0.551348
leis_sports_cnt	-0.174064	1.000000	0.087771
ref_view_cnt	0.551348	0.087771	1.000000

02 AVERAGE MAXIMUM TEMPERATURE (°C)와 타겟 변수들 간의 상관관계



02 AVERAGE MAXIMUM TEMPERATURE (°C) 상관관계 분석 [코드]

상관관계 분석을 해 보니 Average maximum temperature (°C)와
leis_sports_cnt(-0.1)는 관계가 없고,
ref_view_cnt(0.5)하고는 관계가 깊다.

```
df_Average_maximum_temperature = df[["Average maximum
temperature (°C)","leis_sports_cnt","ref_view_cnt"]]
df_Average_maximum_temperature.corr()
```

02 AVERAGE MINIMUM TEMPERATURE (°C) 컬럼 분석

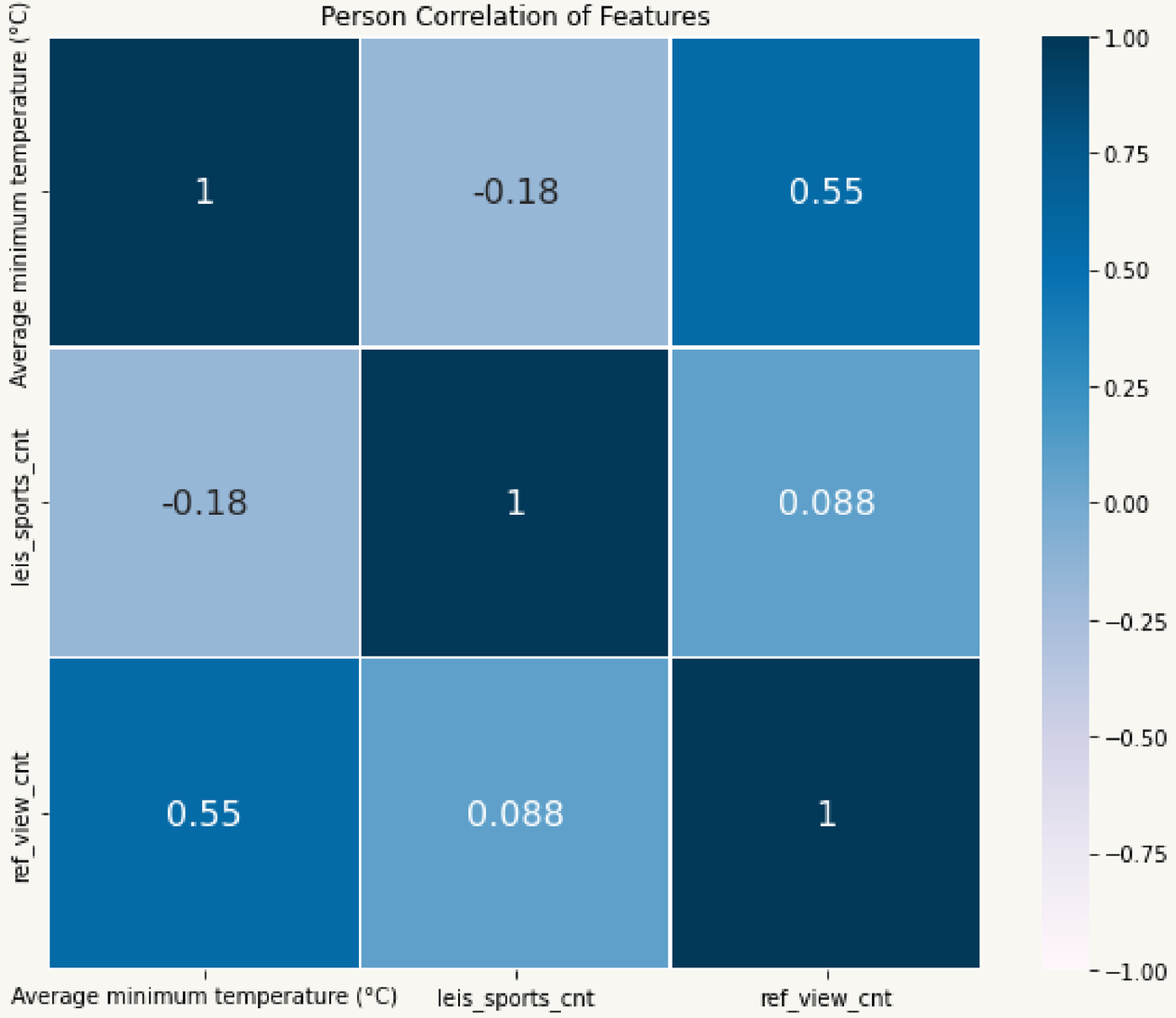
base_month	Average minimum temperature (°C)
201501	4.9
201502	4.7
201503	7.3
201504	12
201505	15.9
201506	19.5
201507	23.2
201508	24.1
201509	20.7
201510	16.3

온도가 높을 수록
여행객이 많이 온다면
온도가 낮을 수록
여행객이 적게 올 것이라
예상했습니다.

02 AVERAGE MINIMUM TEMPERATURE (°C)와 타겟 변수들 간에 상관관계

	Average minimum temperature (°C)	leis_sports_cnt	ref_view_cnt
Average minimum temperature (°C)	1.000000	-0.177321	0.554835
leis_sports_cnt	-0.177321	1.000000	0.087771
ref_view_cnt	0.554835	0.087771	1.000000

02 AVERAGE MINIMUM TEMPERATURE (°C)와 타겟 변수들 간에 상관관계



02 AVERAGE MINIMUM TEMPERATURE (°C) 상관관계 분석 [코드]

상관관계 분석을 해 보니 Average minimum temperature (°C)와
leis_sports_cnt(-0.1)는 상관관계가 없고,
ref_view_cnt(0.5)는 상관관계가 깊다.

```
df_Average_minimum_temperature = df[["Average minimum
temperature (°C)","leis_sports_cnt","ref_view_cnt"]]
df_Average_minimum_temperature.corr()
```

02 MAXIMUM TEMPERATURE (°C) 컬럼 분석

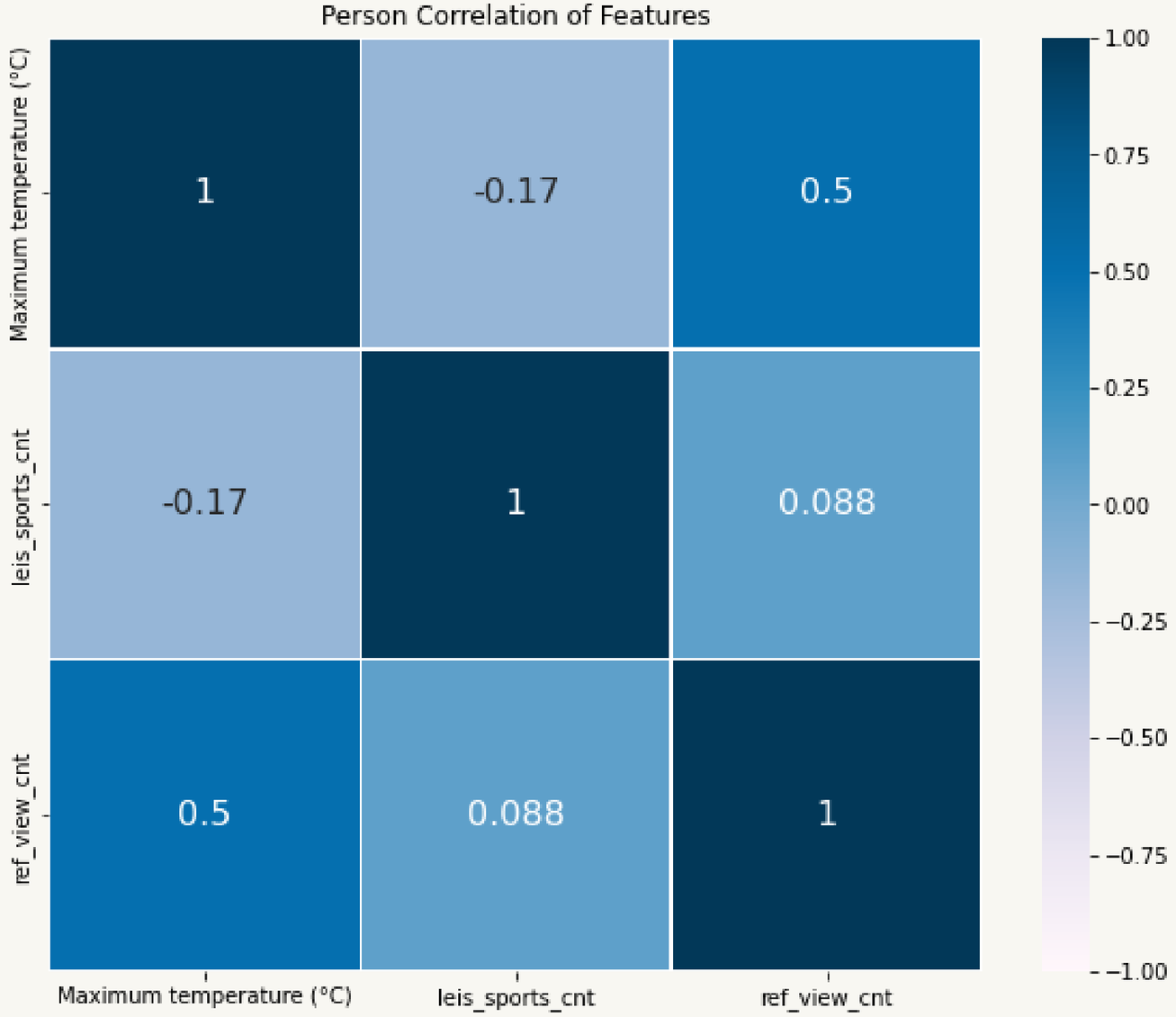
base_month	Maximum temperature (°C)
201501	17.5
201502	17.3
201503	22.2
201504	27.7
201505	29.4
201506	30.6
201507	36.7
201508	35.5
201509	28.6
201510	27.5

기온이 높을 수록
바다를 보러오는 여행객들이
더 많이 올 것이라 예상했습니다.

02 MAXIMUM TEMPERATURE (°C) 와 타겟 변수들 간에 상관관계

index	Maximum temperature (°C)	leis_sports_cnt	ref_view_cnt
Maximum temperature (°C)	1.0	-0.16556108874508063	0.5008026559016912
leis_sports_cnt	-0.16556108874508063	1.0	0.08777131563384108
ref_view_cnt	0.5008026559016912	0.08777131563384108	1.0

02 MAXIMUM TEMPERATURE (°C) 와 타겟 변수들 간에 상관관계



02 MAXIMUM TEMPERATURE (°C) 상관관계 분석 [코드]

상관관계 분석을 해 보니 Maximum temperature (°C)와
leis_sports_cnt(-0.1)는 상관관계가 없고,
ref_view_cnt(0.5)는 상관관계가 깊다.

```
df_Maximum_traturempee = df[["Maximum temperat
ure (°C)","leis_sports_cnt","ref_view_cnt"]]
df_Maximum_traturempee.corr()
```

02 MINIMUM TEMPERATURE (°C) 컬럼 분석

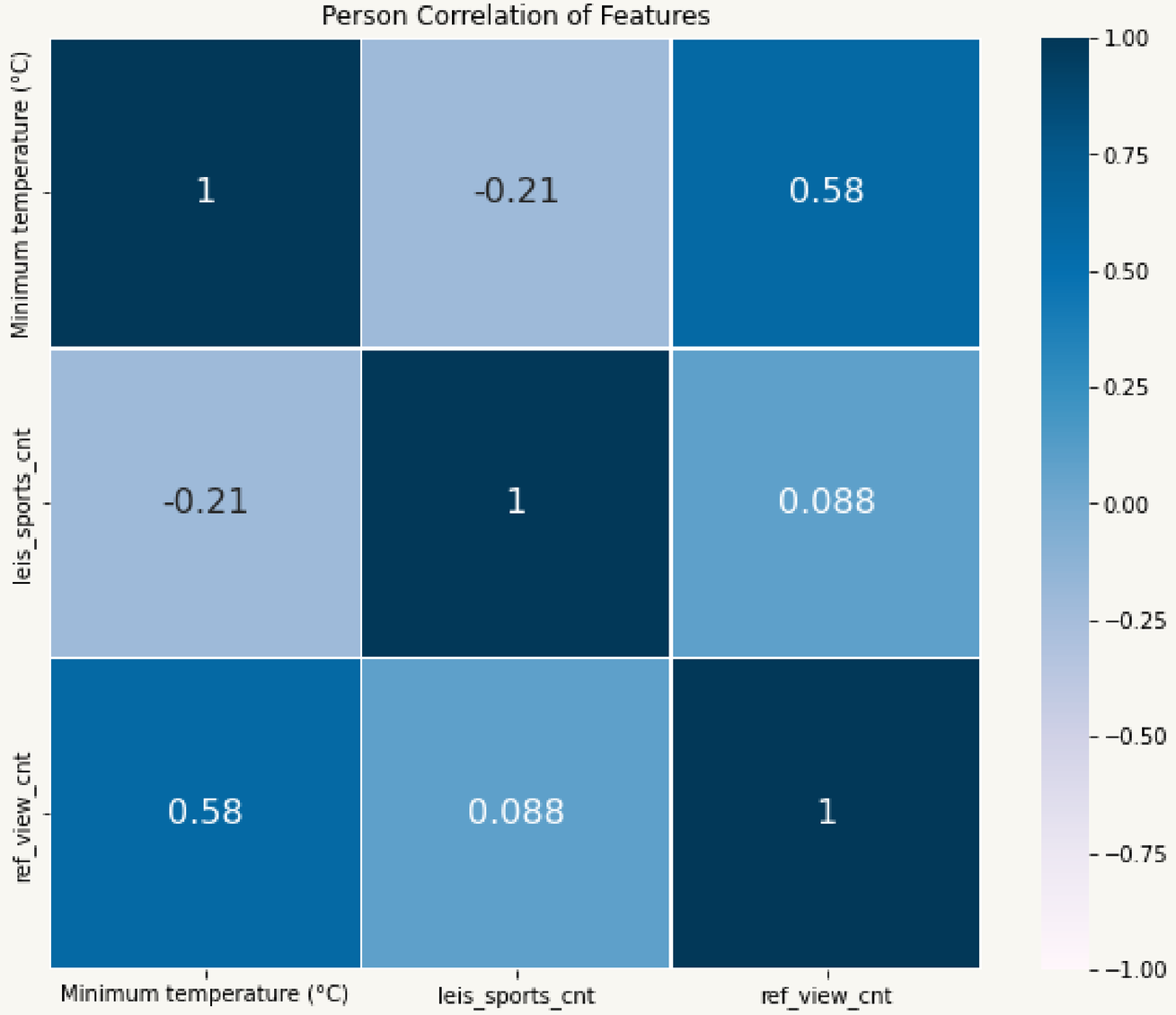
base_month	Minimum temperature (°C)
201501	1.2
201502	-0.7
201503	0.2
201504	7
201505	11
201506	15.3
201507	19.1
201508	20.4
201509	18.4
201510	11.7

온도가 높을 수록
여행객이 많이 온다면
온도가 낮을 수록
적게 올 것이라
예상했습니다.

02 MINIMUM TEMPERATURE (°C)와 타겟 변수들 간에 상관관계

index	Minimum temperature (°C)	leis_sports_cnt	ref_view_cnt
Minimum temperature (°C)	1.0	-0.2071295974921469	0.5825543032947853
leis_sports_cnt	-0.2071295974921469	1.0	0.08777131563384108
ref_view_cnt	0.5825543032947853	0.08777131563384108	1.0

02 MINIMUM TEMPERATURE (°C)와 타겟 변수들 간에 상관관계



02 MINIMUM TEMPERATURE (°C)상관관계 분석 [코드]

상관관계 분석을 해 보니 Minimum temperature (°C)와
leis_sports_cnt(-0.2)는 상관관계가 없고,
ref_view_cnt(0.5)는 상관관계가 깊다.

```
df_Minimum_traturempee = df[["Minimum temperat
ure (°C)","leis_sports_cnt","ref_view_cnt"]]
df_Minimum_traturempee.corr()
```

02 AVERAGE WIND SPEED (M/S) 컬럼 분석

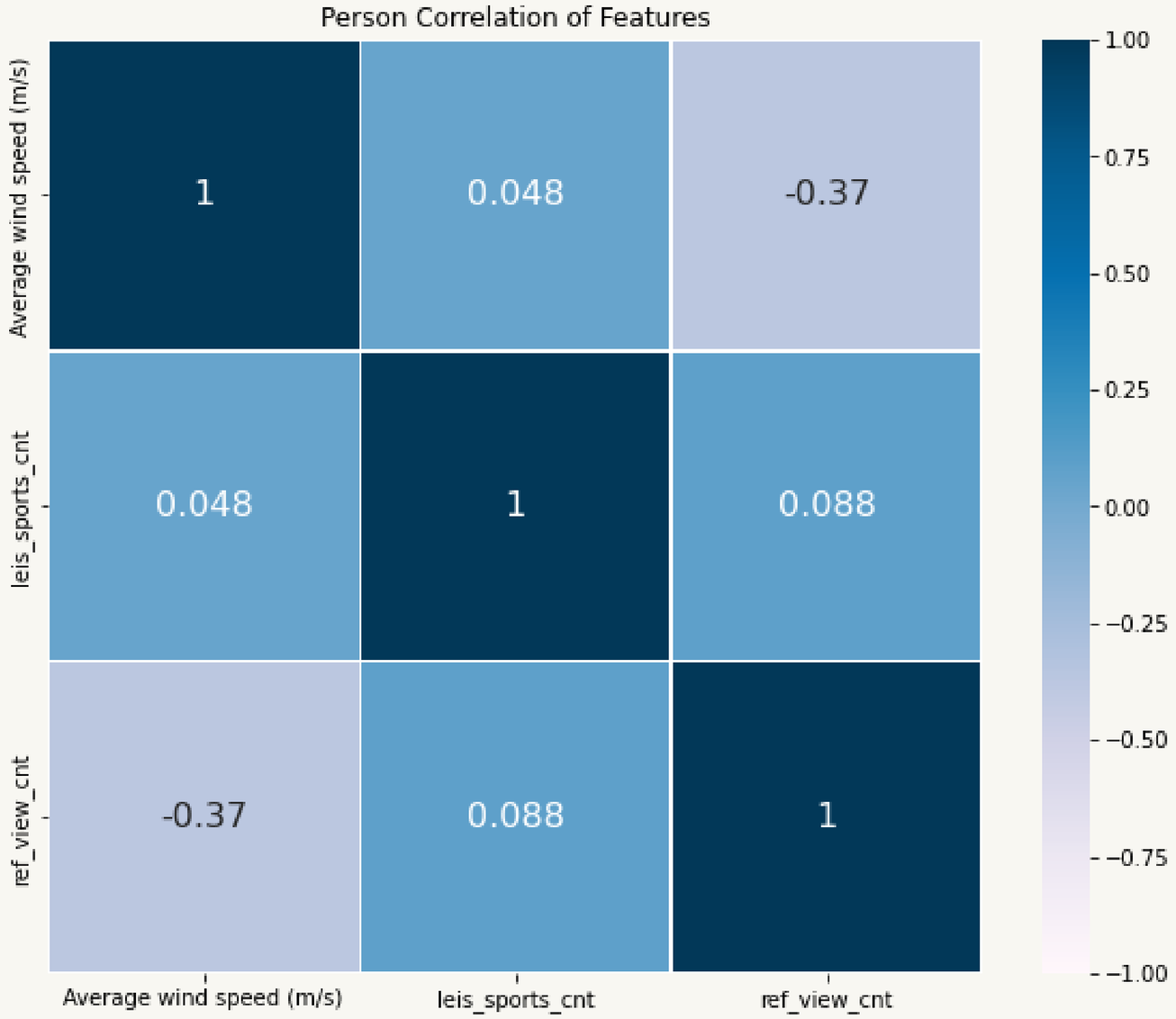
base_month	Average wind speed (m/s)
201501	3.9
201502	3.7
201503	2.9
201504	2.9
201505	2.6
201506	2.3
201507	3.1
201508	2.5
201509	2.7
201510	3.1

제주도는 바람이 많이 불기에
바람이 적게 불 수록
여행객이 많을 거라
예상했습니다.

02 AVERAGE WIND SPEED (M/S)과 타겟 변수들 간에 상관관계

index	Average wind speed (m/s)	leis_sports_cnt	ref_view_cnt
Average wind speed (m/s)	1.0	0.04844866973464912	-0.36786769279701004
leis_sports_cnt	0.04844866973464912	1.0	0.08777131563384108
ref_view_cnt	-0.36786769279701004	0.08777131563384108	1.0

02 AVERAGE WIND SPEED (M/S)과 타겟 변수들 간에 상관관계



02 AVERAGE WIND SPEED (M/S) 상관관계 분석 [코드]

상관관계 분석을 해 보니 Average wind speed (m/s)와
leis_sports_cnt(0.04)는 상관관계가 없고,
ref_view_cnt(-0.3)는 상관관계가 있다.

```
df_Average_wind_speed = df[["Average wind speed
(m/s)","leis_sports_cnt","ref_view_cnt"]]
df_Average_wind_speed.corr()
```

02 REF_VIEW_CNT와 상관관계 계수가 높은 컬럼 TOP.10

컬럼	상관계수
Minimum temperature (°C)	0.582554
평균지면온도(°C)	0.559266
Average minimum temperature (°C)	0.554835
0.3m평균지중온도(°C)	0.554146
최저초상온도(°C)	0.553892
Average temperature (°C)	0.552147
소형총증발량(mm)	0.551732
Average maximum temperature (°C)	0.551348
평균수증기압(hPa)	0.550602
0.05m평균지중온도(°C)	0.550031

상관계수가 높은 컬럼들을 보니
기온에 관련된 컬럼들이 많은 것을 볼 수 있습니다.
그러므로 ref_view_cnt의 값은
기온의 영향을 많이 받는다.

02 LEIS_SPORTS_CNT와 상관관계 계수가 높은 컬럼 TOP.10

컬럼	상관관계
평균현지기압(hPa)	0.226816
평균해면기압(hPa)	0.225545
최고해면기압(hPa)	0.213945
최저해면기압(hPa)	0.212689
최다풍향(16방위)	0.191526
season_autumn	0.159662
최대순간풍속 풍향(16방위)	0.081648
season_spring	0.064811
Average wind speed (m/s)	0.048449
최대풍속 풍향(16방위)	0.033766

leis_sports_cnt와 상관계수가 높은 컬럼들을 보았을 때,
0.2정도인 걸로 보아
지금 가지고 있는 컬럼들은
leis_sports_cnt와 상관계수가 매우 낮고,
예측이 어려울 것으로 예상된다.

03

전처리 과정

03

03 전처리 과정

1) SEASON 컬럼의 원핫인코딩을 진행

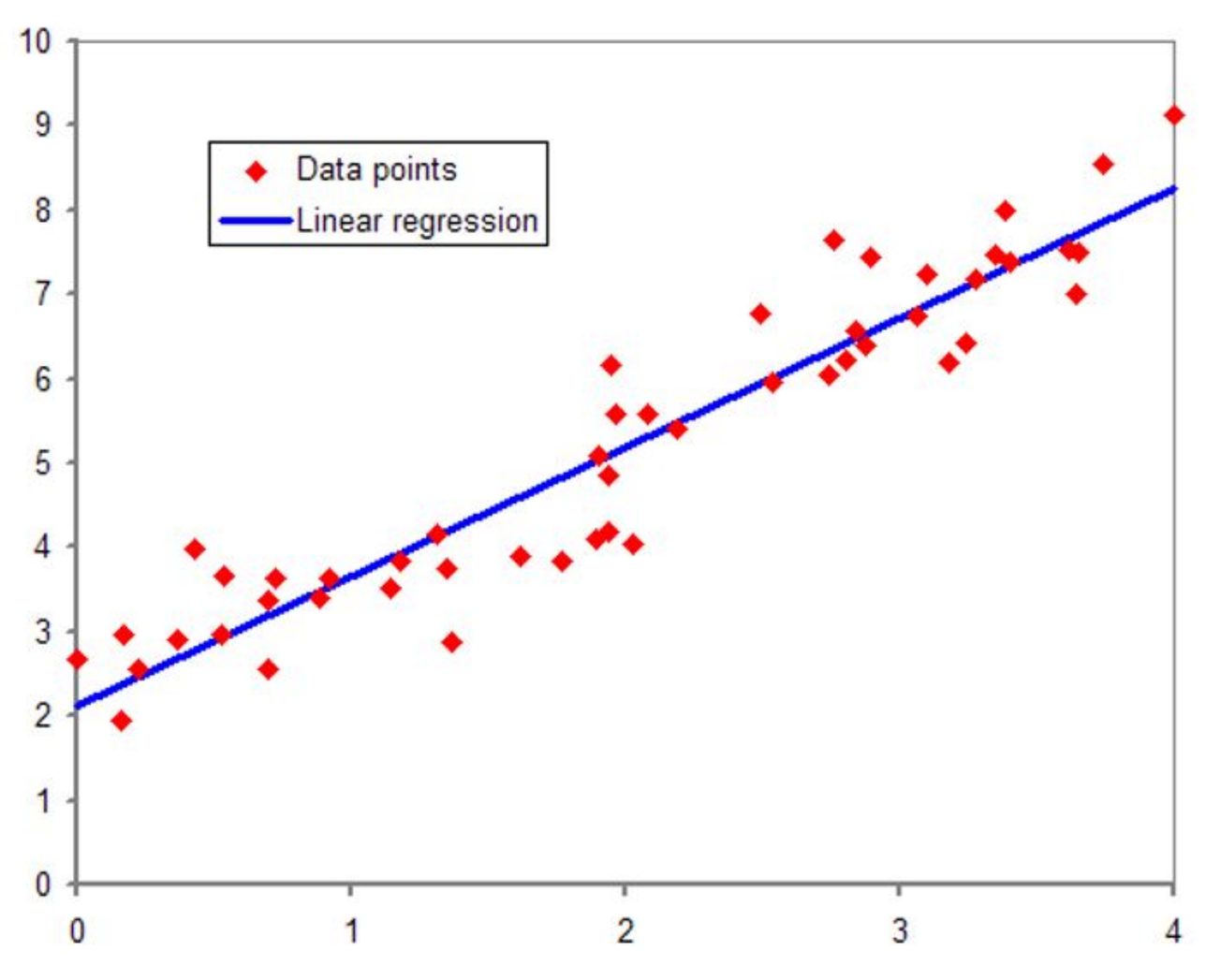
* 이미 데이터 자체를 구글 스프레드 시트를 통해 만들었에
  원핫인코딩을 제외하고는 따로 전처리 과정이 없음.

04

선형회귀 분석

04

04 선형회귀 분석이란??



04 선형회귀 분석이란??

"통계학에서 사용하는 자료 분석 방법 중 하나로,
간략히 표현해 여러 자료들 간의 관계성을
수학적으로 추정, 설명한다."

*자료라 함은 본 대회에서 사용한
외부데이터 61개 컬럼을 의미합니다.

04 선형회귀 분석 [코드]

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# feature, target 데이터 분리
X_data = df.drop(['leis_sports_cnt', 'ref_view_cnt'], axis = 1)
y_target = df['ref_view_cnt']
# train, test 데이터 분리
X_train , X_test , y_train , y_test = train_test_split(X_data , y_target , test_size=0.3,
random_state=156)

# Linear Regression
lr = LinearRegression()
# fit 메소드 학습 : 주어진 데이터로 estimator(사이킷런이 제공) 알고리즘 학습
lr.fit(X_train, y_train)
```

04 선형회귀 분석 [코드]

```
# predict 메소드 : 학습된 모델로 예측을 수행
y_preds = lr.predict(X_test)
y_preds[0:5]

# rmse를 활용한 평가
mse = mean_squared_error(y_test, y_preds)
rmse = np.sqrt(mse)

print(f'MSE : {mse:.3f}, RMSE: {rmse:.3f}')
print(f'Variance score : {r2_score(y_test, y_preds):.3f}')
```

05

최종 예측 결과

05

05 REF_VIEW_CNT의 RMSE 평가 결과

평가지표	값
MSE	17,931,821,106.03
RMSE	133909.75

05 LEIS_SPORTS_CNT의 RMSE 평가 결과

평가지표	값
MSE	3,069,135,749.30
RMSE	55399.781

05 9월달 REF_VIEW_CNT 입도객 수 예측 코드 & 결과

#9월의 내국인 관광목적 예측 입도객 수(ref_view_cnt)

```
lr.predict(df_b.values)
```

컬럼 명	예측 결과
ref_view_cnt	931431

06

한계점

06

06 한계점

1) 온도 관련 컬럼을 여러개가 아닌 한 개로 묶기

2) 상관관계가 낮더라도 다양한 컬럼을 추가하기

3) 중요컬럼을 뽑을 때 음수 값도 넣기

4) 코로나 데이터 추가하기

5) 환율 데이터 추가하기

감사합니다

지금까지 유지민이었습니다.

uujiwino598@naver.com
(깃허브 계정)