데이터 사이언스
term project 보고서

-32230811 김수인

순서

[요약]

분석주제	바다 네비게이션을 통해 해양선박 사고에 영향을 주는 구체적인 요인을 분석할 수 있을까?
데이터셋	해양수산부_바다내비_사고상황 마스터, 해양수산부_바다내비_사고상황 선박정보
분석도구	평균, 표준편차, 막대그래프, 선그래프, 산점도, 상관계수, 히트맵, 정규성검정
분석내용 요약	1-1 -(막대그래프) 사고발생 빈도수: 선박 간 교차>선박 간 대립> 암초 -(막대그래프) 7-11시 사이에 사고가 가장 많이 발생했다. -(산점도) 최근에 올수록 사고율이 줄어들었다. -(상관계수)사고유형과 사고일시의 상관계수는 매우 낮다. -(히트맵) 사고유형과 사고일시의 상관계수를 히트맵으로 나타냈다. -(정규성검정) 사고일시의 정규성 검정: Shapiro-Wilk 테스트는 정규분포를 따른다/ p-value는 정규분포를 따르지 않는다. -(Q-Q플랏) 사고일시는 정규분포를 따른다고 볼 수 없다. 1-2 -(산점도) 대지속력과 대지첨로간의 명확한 상관관계는 보이지 않는다. -(히트맵) 대지속력과 대지첨로간의 상관관계는 낮다. -(선그래프) 4월, 8월, 12월에 사고가 일어난 선박들은 다른 달보다 높은 속력을 유지했다. -(정규성검정) 대지속력의 Shapiro-Wilk 테스트, p-value둘 다 정규성을 따른다. -(Q-Q플랏) 대지속력은 정규성을 크게 벗어난다. -(정규성검정) 대지침로의 Shapiro-Wilk 테스트, p-value둘 다 정규성을 따른다. -(Q-Q플랏) 대지침로는 정규성을 크게 벗어난다.
결론 요약	이 보고서는 바다 네비게이션 데이터를 활용해 해양 선박 사고의 주요 요인들을 본석했으며, 이러한 결과들은 해양선박 안전시스템에 기초자료가 되어서 자주 발생하는 시간대와 원인을 반영하여 사고 예방을 위한 안전 시스템망 구축에 중요한 역할을 할 것이다.
이 분석의 가치	준비된 데이터가 아닌 직접 데이터를 선별해 분석해보니 생각만큼 쉽지 않다는 것을 느꼈다. 일단 수업에서 할 수 있는 것은 다 해보려 노력했지만 이 데이터에 맞는 분석방법은 제한적이라는 것을 깨달았다. 특히, 수업때 정말 재미있게 배웠던 Map을 활용해 지도위에 사고가 난 지역을 표시하고 싶었지만 코드를 직접 작성해보니, 모든 데이터의 위도와 경도의 차이가 미미해 명확히 구분하기 어려워서 제외시킨것이 너무 아쉬웠다. 이 분석은 사고가 일어나는 원인과 시간대, 기간을 알 수 있다는 것이며 이는 바다 위에서 일어나는 사고들의 유형을 파악과 원인을 파악하는데 도움을 줄 수 있을것이

	다. 또한, 기초적인 코드를 이용해 작성했기에 코딩기초를 배우는 사람들이 이 분석을 참고해 직접 작성해보면 내가 그랬듯이 재미를 느낄 수 있을거라 생각한다. 처음하는 데이터 분석이라서 장점보다는 안맞는부분이 많고 부족한부분이 매우 많지만, 배운것을 적용해 도움없이 혼자 해보는 첫 프로젝트라는 점에서 이 보고서는 나에게 엄청난 가치를 지닌다. 분석을 하며 처음으로 코딩을 배우길 잘했다는 생각이 들정도였다. 후에 학년이 올라가 더 어려운 코드들을 배우게된다면 이 보고서를 다시 분석해보고 싶다.

[서론]

기말 term프로젝트의 주제로 어떤 데이터셋을 사용할지 고민하던 중, 바다 위에는 선박의 다양한 정보를 제공해주는 바다네비게이션이 있다는 것을 처음 알게되었다. 바다 위는 땅처럼 고정되어 있지 않고, 계속해서 변화하는 곳이기에 이런곳에서 사고가 난다면 바다 네비게이션으로 어떤것을 알 수 있을까? 라는 궁금증도 생겼다. 따라서, 바다 위에서 일어나는 사고를 바다 네비게이션을 활용해 분석해보면 좋겠다는 생각이 들어 주제를 선정했다.

[본론]

★데이터 소개

사용한 데이터셋 2가지는 모두 해양수산부에서 제공한 것이다. 두 데이터셋 모두 행이 약 9000여개 정도로 분석하기에 부족하지 않은 표본이라 판단했다.

1-1데이터에서는 대상선박 대지속력, 대상선박 대지침로, 등록일시가 유의미한 컬럼이라고 생각했다. 그 이유는, 대상선박의 위도와, 경도는 모든 샘플들이 미세한 차이만 있을뿐 다 비슷한 값을 가지고 있었고 사고상황과 선박 통신만 컬럼또한 모든값이 일치해 데이터 분석을 하기에 큰 가치가 없다고 판단했다.

1-2데이터에서는 사고일시, 사고구분, 사고유형, 등록일시가 데이터를 분석하기에 유의미한 컬럼이라고 판단했다.

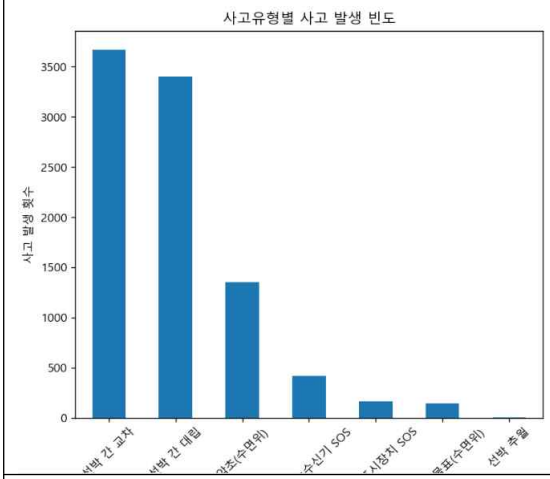
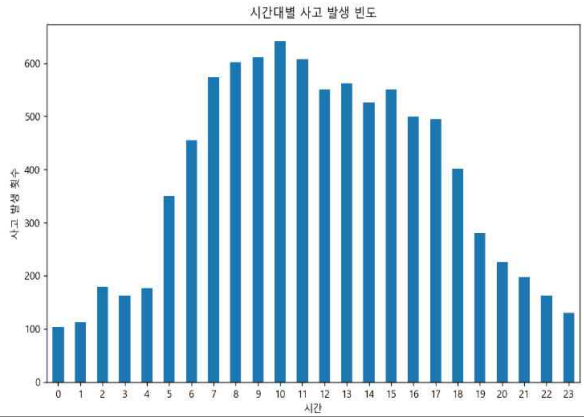
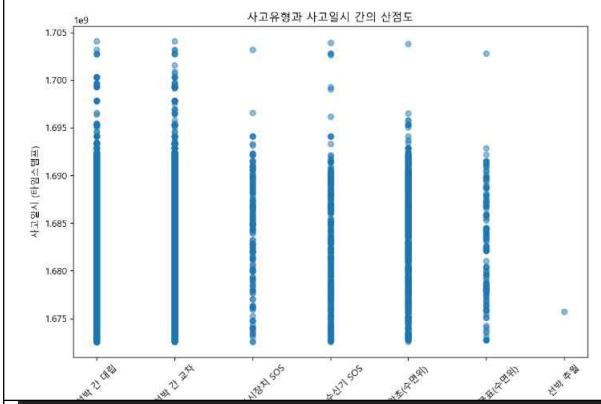
	A	B	C	D	E	F	G	H
1	선박 통신(	대상선박 ㄷ	대상선박 ㄷ	대상선박 위도	대상선박 경도	사고상황 확인응(	등록일시	수정일시
2	LTE-M	1	117.8	34.22599667	127.0219133	정상응답	2023-01-06 5:11	2023-01-06 5:11
3	LTE-M	3.6	124.2	34.83703833	128.7315567	정상응답	2023-01-08 7:24	2023-01-08 7:24
4	LTE-M	2	277.5	36.316595	129.4055817	정상응답	2023-01-31 3:32	2023-01-31 3:32
5	LTE-M	1.6	334.7	35.48940167	129.40306	정상응답	2023-01-31 15:34	2023-01-31 15:34
6	LTE-M	6.4	169	34.645735	128.4652317	정상응답	2023-01-01 9:41	2023-01-01 9:41

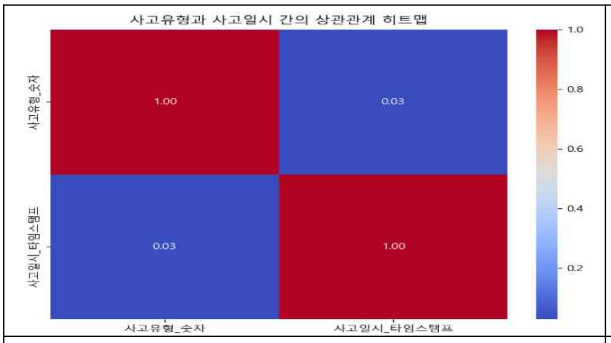
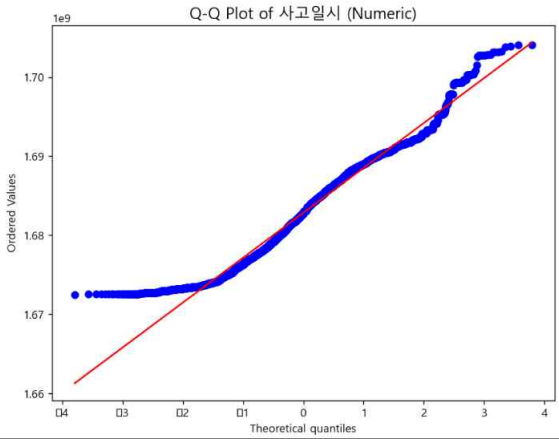
(1-1 바다 내비게이션 선박 정보 데이터셋)

	A	B	C	D	E	F	G	H	I	J
1	사고일시	사고구분	사고유형	사고위치 위도	사고위치 경도	사고상황 ?	사고상황 ?	진행상태 갱신 시	등록일시	수정일시
2	2023-01-01 0:07	충돌	선박 간 대	36.96517039	124.5675214	종결	전파완료여	2023-01-01 0:09	2023-01-01 0:08	2023-01-01 0:09
3	2023-01-01 1:05	충돌	선박 간 대	34.8619784	127.7291478	종결	전파완료여	2023-01-01 1:07	2023-01-01 1:05	2023-01-01 1:07
4	2023-01-01 2:22	충돌	선박 간 대	35.36779321	129.5850326	종결	전파완료여	2023-01-01 2:23	2023-01-01 2:22	2023-01-01 2:23
5	2023-01-01 2:46	충돌	선박 간 대	35.38438603	129.5810551	종결	전파완료여	2023-01-01 2:48	2023-01-01 2:47	2023-01-01 2:48

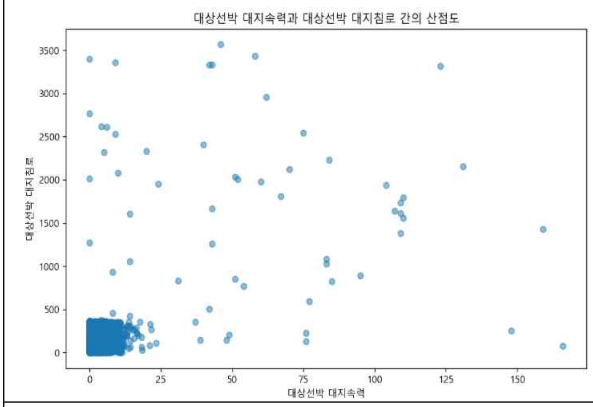
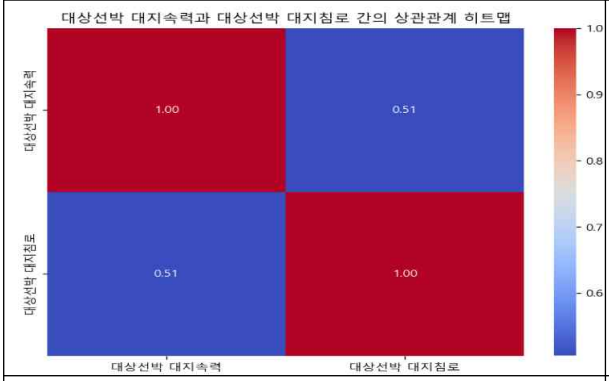
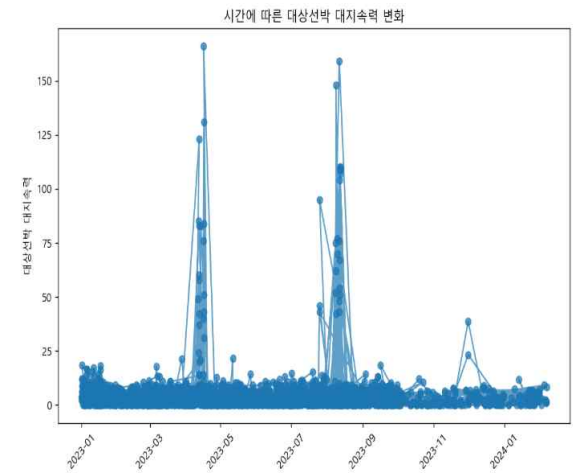
(1-2 바다 네비게아션 사고상황 데이터셋)

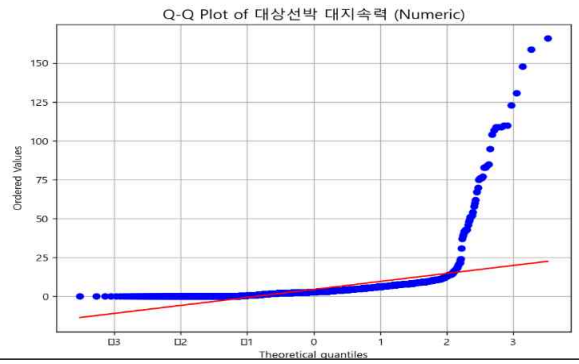
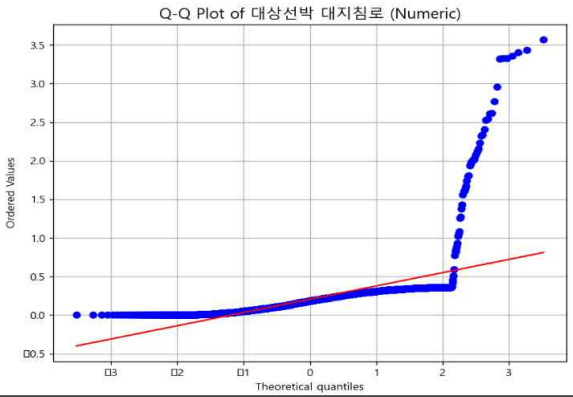
★1-1데이터 분석

분석 결과	해석
	사고유형별 사고발생 빈도수를 플랏으로 나타낸 결과, 1. 선박 간 교차 2. 선박 간 대립 3. 암초(수면 위) 순으로 가장 많이 일어난 것을 알 수 있었다. 특히, 1과 2를 합친것은 전체표본(9156)의 72%에 해당하는 값으로 사고발생의 대부분을 차지한다.
	사고가 많이 발생하는 시간대를 알아보기 위해 사고일시와 사고시간을 플랏으로 나타내어본 결과, 7시-11시 사이에 많이 발생한다는 것을 알 수 있었다. 오히려, 밤에 사고가 많이 일어날 것이라는 추측과 달리 야간사고 발생 빈도가 제일 낮게 나온것이 의외였다.
	사고유형과 사고일시를 효과적으로 시각화하기 위해 산점도 나타낸 결과, 모든 사고가 최근에 올수록 눈의 띄게 줄어든 것을 확인할 수 있었다. 특히, SOS와 관련된 사고들이 줄었으며 이는 긴급신고를 받았을 때, 빠르게 선박의 위치를 정확히 확인하여 구출하는 시스템이 구축되어왔다고 볼 수 있다.
	사고유형과 사고일시의 상관계수를 분석해보았다. 0.029라는 낮은 숫자로, 이 둘 사이에 뚜렷한 연관성이 없다고 볼 수 있다.

	위의 상관계수에 대한 상관관계를 히트맵을 사용하여 시각화 한 것이다. 숫자가 작아질수록 파란색을 띄게 설정했으므로, 둘 사이의 상관관계가 매우 낮은 수치를 띈다는 것을 알 수 있었다.
Shapiro-Wilk Test Statistics: 0.9757705091065839 P-value: 1.1112873019937113e-36	사고일시가 정규성을 따르는지 알아보기 위해 Shapiro-Wilk 테스트를 진행한 결과, 정규성의 정도를 나타내는 통계량인 Shapiro-Wilk Test Statistic의 값이 1과 비슷하므로 이 테스트에선 정규성을 띈다고 해석할 수 있었다. 다만, P-value값은 0.05(유의수준)보다 매우 작은값을 띄므로 귀무가설을 기각하고 정규분포를 따르지 않는다고 해석된다. 이 두가지 테스트의 결과가 반대인 이유는 샘플크기의 영향일수도 있고, 작은 차이에도 민감하게 반응하는 p-value때문일 수도 있다, 따라서, Q-Q플랏으로 정규성을 확인해 보기로했다.
	파란점들은 데이터의 실제 분포를 나타내고, 빨간선은 정규분포를 나타낸다. 데이터에서 왼쪽 꼬리부분을 제외하고는 빨간석을 따르고 있으므로, 이부분은 정규분포를 따른다고 볼 수 있지만, 전체적으로 보면 꼬리부분이 정규분포에서 많이 벗어나 있으므로 정규분포를 따른다고 볼 수 없다.

★1-2데이터 분석

분석 결과	해석
LTE-M 4.389703	대상선박 대지속력의 평균값이다.
	x축은ㅇ 대상선박의 속력을, y축은 대상선박의 진행방향을 나타낸다. 대부분의 데이터가 특정 구간(0-25,0-300)에 밀집되어 있으며 이는 대부분의 선박이 낮은 속도로 움직이고 대체로 일정한 범위내에서 방향을 유지하고 있음을 의미한다. 이 산점도로는 속도와 방향간의 명확한 상관관계는 보이지 않았다.
	대지속력과 대지침로간의 상관관계를 분석해본 결과, 0.51이라는 낮은 숫자로 상관관계가 있다고 보기에는 어려운 숫자가 나왔다.
	시간에 따른 대상선박 대지속력의 변화를 분석해본 결과, 4월, 8월, 12월에 다른 달보다 높은 속력을 유지했다는 것을 알 수 있었다. 이 이유를 찾아보니 운송수요와 관련있는것을 알 수 있었다. 평균적으로 우리나라는 4월에는 제조업 물품수송이 증가하고, 8월에는 휴가철과 관련된 관광물품이, 12월에는 연말 물류와 선물운송이 증가한다. 이러한 수요증가가 선박이 더 높은 속도로 운영하도록 했을것이다.
대상선박 대지속력 Shapiro-Wilk Test Statistics: 0.2795271942757O1 대상선박 대지속력 P-value: 3.770300177088378e-77	Shapiro-Wilk 테스트의 경우, 정규분포에서 크게 벗어났다고 해석할 수 있으며, p-value도 매우 작은값으로 귀무가설이 기각되며, 데이터가 정규분포를 따르지 않는다고 해석 가능하다.

	정규성 테스트로 예상했듯이, 대지속력의 정규성을 시각화해주는 Q-Q플랏으로 그려본 결과, 정규성을 크게 벗어났음을 시각화를 통해 확인 할 수 있었다.
대상선박 대지침로 Shapiro-Wilk Test Statistics: 0.42481007124667003 대상선박 대지침로 P-value: 1.596272074681356e-72	대지속력보다는 높은 값을 띄지만 위와 마찬가지로 두 테스트 다 정규분포에서 크게 벗어났다고 해석할 수 있었다.
	대지침로의 정규성을 시각화해주는 Q-Q플랏 또한 빨간선에서 크게 벗어났으므로, 정규성을 따르지 않는것을 확인 할 수 있었다.

[결론]

본 데이터 분석 프로젝트에서는 바다네비게이션 데이터를 활영해 해양 선박 사고의 주요 요인들을 분석했다. 다양한 분석과 시각화를 통해 1-2데이터셋에서는 발생 빈도, 사고 시간대, 사고 유형간의 관계등을 파악할 수 있었다. 첫째, 사고유형의 경우, 선박간 교차와 대립, 그리고 암초충돌이 전체의 72%를 차지했다. 둘째, 사고발생시간대를 통해 사고는 주로 오전 7-11시 사이에 집중된다는 사실을 발견할 수 있었다. 야간 사고 발생빈도가 매우 낮다는 점에서 깨달음을 주는 결과였다. 셋째, 사고유형과 사고일시간에는 유의미한 상관관계가 없었으며, 이는 두 변수간에 직접적인 연관성이 없음을 뜻한다. 1-2데이터셋에서는 첫째, 선박의 속도와 진행방향간에는 명확한 상관관계가 없음을 알 수 있었다. 둘째, 운송수요가 증가하는 특정시기(4월, 8월, 12월)에는 속도증가로 인한 사고가 많이 발생함을 알 수 있었다. 마지막으로 대지속력과 대지침로는 모두 정규분포를 따르지 않음을 확인할 수 있었다.

이번 분석을 통해 바다 네비게이션 데이터를 활용한 사고분석이 사고 발생의 주요 패턴

을 파악하는데 유용하다는 것을 알 수 있었다. 이 분석결과는 향후 해양선박 안전관리 시스템 개선에 기초자료역할을 할 수 있을것이라 기대되며 특히, 사고가 많이 일어나는 시간대와 기간, 그리고 사고발생원인등을 고려해 안전시스템망을 구축하면 앞으로 일어나는 바다 위의 사고를 예방 할 수 있을것이라 확신한다.

[별첨-사용한 코드]

- 1-1데이터셋

```
#한글 글꼴 설정
plt.rcParams['font.family'] = 'Malgun Gothic'
# 데이터 불러오기
data = pd.read_csv( filepath_or_buffer: "C:\\Users\\User\\OneDrive\\바탕 화면\\해양수산부_바다내비_사고상황 마스터.CSV",
                   encoding='euc-kr')

# 사고유형별 사고 빈도 계산
accident_type_counts = data['사고유형'].value_counts()

# 시각화
accident_type_counts.plot(kind='bar', figsize=(7, 6))
plt.title('사고유형별 사고 발생 빈도')
plt.xlabel('사고유형')
plt.ylabel('사고 발생 횟수')
plt.xticks(rotation=45)
plt.show()
```

->사고빈도 막대그래프

```
#2. 사고발생 시간 분석#############################################################
# 사고일시를 datetime 형식으로 변환
data['사고일시'] = pd.to_datetime(data['사고일시'])

# 사고 발생 시간 추출 (시간만)
data['사고시간'] = data['사고일시'].dt.hour

# 시간대별 사고 발생 빈도 계산
accident_time_counts = data['사고시간'].value_counts().sort_index()

# 시각화
accident_time_counts.plot(kind='bar', figsize=(10, 6))
plt.title('시간대별 사고 발생 빈도')
plt.xlabel('시간')
plt.ylabel('사고 발생 횟수')
plt.xticks(rotation=0)
plt.show()
```

->사고일시 막대그래프

```python
import pandas as pd
import matplotlib.pyplot as plt

# '사고일시' 열을 datetime 형식으로 변환
data['사고일시'] = pd.to_datetime(data['사고일시'], errors='coerce')

# 산점도 그리기
plt.figure(figsize=(10, 6))

# 사고유형을 텍스트로 표시
plt.scatter(data['사고유형'], data['사고일시'], alpha=0.5)
plt.title('사고유형과 사고일시 간의 산점도')
plt.xlabel('사고유형')
plt.ylabel('사고일시 (타임스탬프)')
plt.xticks(rotation=45)  # X축 레이블 각도 조정
plt.gca().yaxis_date()
plt.show()
```

->사고유형과 사고일시의 산점도

```python
# 상관관계 계산
import pandas as pd

# 사고유형을 숫자로 변환 (범주형 데이터를 숫자로 변환)
data['사고유형_숫자'] = data['사고유형'].astype('category').cat.codes

# 사고일시를 타임스탬프 형식으로 변환 (시간 데이터를 숫자로 변환)
data['사고일시_타임스탬프'] = data['사고일시'].map(lambda x: x.timestamp() if pd.notnull(x) else None)

# 상관계수 계산
correlation = data[['사고유형_숫자', '사고일시_타임스탬프']].corr()
print(correlation)
######################
import seaborn as sns
import matplotlib.pyplot as plt

# 상관계수 계산
correlation_matrix = data[['사고유형_숫자', '사고일시_타임스탬프']].corr()

# 히트맵 시각화
plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f')
plt.title('사고유형과 사고일시 간의 상관관계 히트맵')
plt.show()
```

->사고유형과 사고일시의 상관관계 분석, 히트맵 분석

```python
from scipy import stats
# 사고일시 컬럼을 datetime 형식으로 변환
data['사고일시'] = pd.to_datetime(data['사고일시'], errors='coerce')  # 오류 발생시 NaT 처리

# 타임스탬프 형식으로 변환
data['사고일시_timestamp'] = data['사고일시'].apply(lambda x: x.timestamp() if pd.notnull(x) else None)

# 정규성 검정
stat, p_value = stats.shapiro(data['사고일시_timestamp'].dropna())  # NaN 제거 후 정규성 검정
print(f"Shapiro-Wilk Test Statistics: {stat}")
print(f"P-value: {p_value}")
```

->사고일시 정규성 검정

```python
# 사고일시 컬럼을 직접 숫자형으로 변환하여 Q-Q 플롯 분석
data['사고일시'] = pd.to_datetime(data['사고일시'])  # 사고일시가 시간 형식이라면 변환
data['사고일시_numeric'] = data['사고일시'].astype(np.int64) / 1e9  # 초 단위로 변환

# Q-Q 플롯 생성
fig, ax = plt.subplots(figsize=(8, 6))
stats.probplot(data['사고일시_numeric'], dist="norm", plot=ax)

# 제목 추가
ax.set_title( label: 'Q-Q Plot of 사고일시 (Numeric)', fontsize=14)
plt.show()
```

->사고일시 Q-Q플랏

- 1-2데이터셋

```python
# 데이터 불러오기
data2 = pd.read_csv( filepath_or_buffer: "C:\\Users\\User\\OneDrive\\바탕 화면\\해양수산부_바다내비_사고상황 선박정보.CSV"
                    encoding='euc-kr')

# 선박 통신망 타입별 대상선박 대지속력 평균 계산
mean_speed_by_network = data2.groupby('선박 통신망 타입')['대상선박 대지속력'].mean()
print(mean_speed_by_network)
#####################################
# 산점도 그리기
plt.figure(figsize=(10, 6))
plt.scatter(data2['대상선박 대지속력'], data2['대상선박 대지침로'], alpha=0.5)
plt.title('대상선박 대지속력과 대상선박 대지침로 간의 산점도')
plt.xlabel('대상선박 대지속력')
plt.ylabel('대상선박 대지침로')
plt.show()
```

->대지속력 평균 계산, 대지속력과 대지침로간의 산점도 분석

```
# 상관계수 계산
correlation_matrix = data2[['대상선박 대지속력', '대상선박 대지침로']].corr()

# 히트맵 시각화
plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f')
plt.title('대상선박 대지속력과 대상선박 대지침로 간의 상관관계 히트맵')
plt.show()
```

->대지속력과 대지침로간의 상관계수계산, 히트맵 분석

```
# '등록일시'를 datetime 형식으로 변환
data2['등록일시'] = pd.to_datetime(data2['등록일시'], errors='coerce')

# 시간에 따른 대상선박 대지속력 변화 선그래프
plt.figure(figsize=(10, 6))
plt.plot( *args: data2['등록일시'], data2['대상선박 대지속력'], marker='o', linestyle='-', alpha=0.7)
plt.title('시간에 따른 대상선박 대지속력 변화')
plt.xlabel('등록일시')
plt.ylabel('대상선박 대지속력')
plt.xticks(rotation=45)
plt.show()
```

->등록일시와 대지속력간의 선그래프

```
stat1, p_value1 = stats.shapiro(data2['대상선박 대지속력'].dropna())  # NaN 제거 후 정규성 검정
stat2, p_value2 = stats.shapiro(data2['대상선박 대지침로'].dropna())  # NaN 제거 후 정규성 검정

print(f"대상선박 대지속력 Shapiro-Wilk Test Statistics: {stat1}")
print(f"대상선박 대지속력 P-value: {p_value1}")
print(f"대상선박 대지침로 Shapiro-Wilk Test Statistics: {stat2}")
print(f"대상선박 대지침로 P-value: {p_value2}")
```

->대지속력과 대지침로의 정규성 검정

```
variables = ['대상선박 대지속력','대상선박 대지침로']

for variable in variables:
    plt.figure(figsize=(8, 6))
    stats.probplot(data2[variable], dist="norm", plot=plt)
    plt.title( label: f'Q-Q Plot of {variable} (Numeric)', fontsize=14)
    plt.grid(True)
    plt.show()
```

->대지속력과 대지침로의 Q-Q플랏