경주시 화재발생 건수 및 예측

컴퓨터공학과 오아연

2022년 12월 14일

1. 분석 배경 및 분석 목적

1) 데이터 출처:

A. 공공데이터포털 소방청 화재발생 주소정보 파일데이터

B. 기상청 기상자료개방포털 종관기상관측(ASOS) 파일셋

2) 화재발생 데이터와 종관기상관측 데이터를 이용하여 자료분석 및 시각화, 기계학습을 하고자 함

- 화재유형 데이터 시각화
- 발화요인 분류에 따른 시각화
- 월별화재건수 시각화
- 온습도와 화재발생과의 상관계수 시각화
- 종관기상관측 데이터를 바탕으로 화재발생 예측

2. 데이터 소개 및 분석 방법

1) 데이터 소개

A. 화재발생 주소정보 데이터에는 2018-01-01일부터 2020-12-31일까지의 3년간의 화재발생 데이터로

- 총 21개의 변수
- 121,100행으로 구성

변수명(컬럼명)	자료설명
연번	화재발생 데이터에 대한 일렬 번호
사망	사망자 수
부상	부상자 수
인명피해(명)소계	사망자 수와 부상자 수의 합
재산피해소계	화재발생으로 인한 총 재산피해액
화재발생년월일	화재가 발생한 날짜(YYYY-MM-DD)와 시간(HH:MM)
시도	전국 행정 구역의 시, 도
시군구	전국 기초자치단체의 시, 군, 구
읍면동	전국 기초자치단체의 읍, 면, 동
화재유형	화재유형
발화열원	발화의 최초원인이 된 불꽃
발화열원소분류	발화열원 소분류
발화요인대분류	발화요인 대분류
발화요인소분류	발화요인 소분류
최초착화물대분류	발화열원에 의해 최초로 불이 붙은 가연물의 대분류
최초착화물소분류	발화열원에 의해 최초로 불이 붙은 가연물의 소분류
장소대분류	장소 대분류
장소중분류	장소 중분류
장소소분류	장소 소분류

B. 종관기상관측(ASOS) 파일셋에는 2018-01-01일부터 2021-12-31일까지의 4년간의 관측 데이터로

- 총 27개의 변수
- 35,064행으로 구성

변수명(컬럼명)	자료설명
지점	지점
일시	일시
기온(°C)	기온(°C)
강수량(mm)	강수량(mm)
풍속(m/s)	풍속(m/s)
풍향(16방위)	풍향(16방위)
습도(%)	습도(%)
증기압(hPa)	증기압(hPa)
이슬점온도(°C)	이슬점 온도(°C)
현지기압(hPa)	현지기압(hPa)
해면기압(hPa)	해면기압(hPa)
일조(hr)	일조(hr)
일사(MJ/m2)	일사(MJ/m2)
적설(cm)	적설(cm)
3시간신적설(cm)	3시간 신적설(cm)
전운량(10분위)	전운량(10분위)
중하층운량(10분위)	중하층 운량(10분위)
운형(운형약어)	운형(운형약어)
최저운고(100m)	최저운고(100m)
시정(10m)	시정(10m)
지면상태(지면상태코드)	지면상태(지면상태코드)
현상번호(국내식)	현상번호(국내식)
지면온도(°C)	지면온도(°C)
5cm 지중온도(°C)	5cm 지중온도(°C)
10cm 지중온도(°C)	10cm 지중온도(°C)
20cm 지중온도(°C)	20cm 지중온도(°C)
30cm 지중온도(°C)	30cm 지중온도(°C)

2) 분석 방법

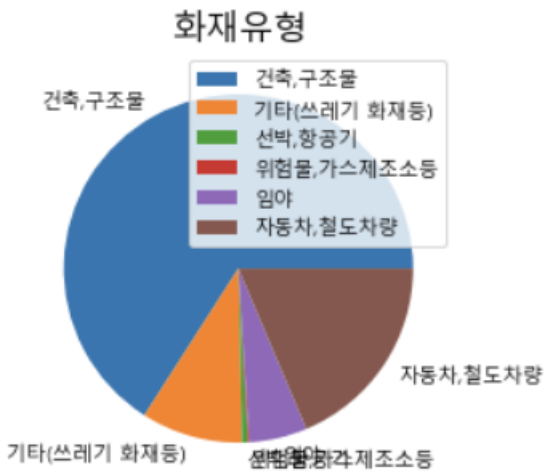
A. 종관기상관측 데이터를 바탕으로 화재발생 예측에 로지스틱 회귀분석을 사용[1]

3. 분석 결과

3.1 화재유형 데이터 시각화

1) 화재발생 당시 화재유형은 아래와 같으며 다음과 같이 분류됨

● 화재유형 중 '건축,구조물'이 상당 부분을 차지하며 '위험물,가스제조소등'이 가장 적은 비율을 차지함

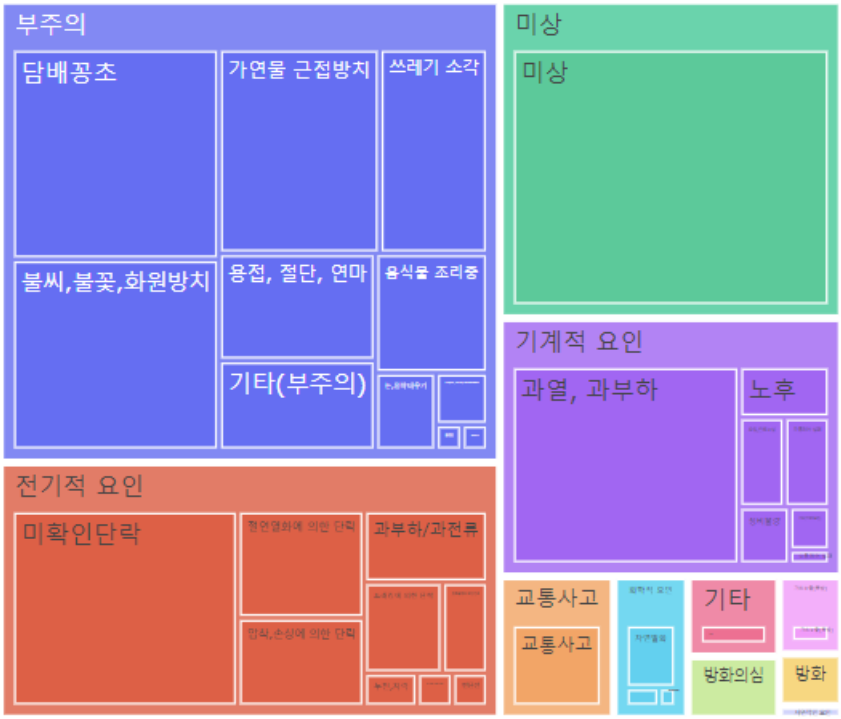


3.2 발화요인 분류에 따른 시각화

2) 화재발생 당시 발화요인은 아래와 같으며, 특징은 다음과 같음

● 화재발생 당시 발화요인은 부주의가 제일 많았으며, 전기적 요인 및 원인 미상이 그 뒤를 이었음.

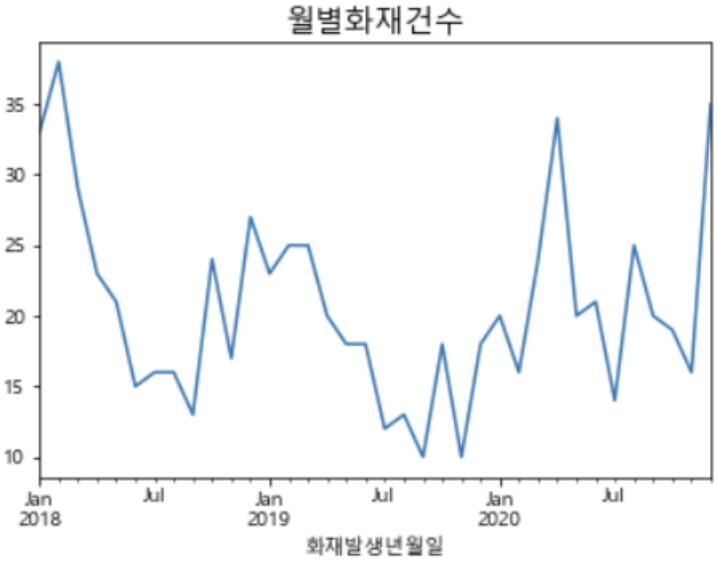
● 당초 목표였던 날씨에 의한 화재 여부에는 크게 영향을 미치지 못할 것으로 파악되었으나, 상관 계수 및 회귀 분석을 진행하며 결론을 이끌어내기로 하였음.


부주의
담배꽁초
가연물 근접방치
쓰레기 소각
불씨,불꽃,화원방치
용접, 절단, 연마
용식물 조리중
기타(부주의)
전기적 요인
미확인단락
정전열화에 의한 단락
과부하/과전류
압착,손상에 의한 단락
미상
미상
기계적 요인
과열, 과부하
노후
교통사고
교통사고
기타
방화의심
방화

3.3 월별화재건수 시각화

3) 연월일에 따른 화재발생 건수는 아래와 같음

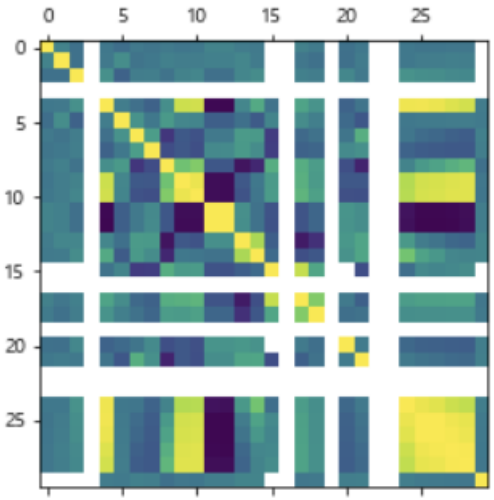
- 화재발생 건수는 2018년 2월이 가장 많고 2019년 9월과 11월이 가장 적음을 보임
- 이에 비추어 보았을 때, 4계절 중 봄, 겨울이 화재 건수가 특히 높은 것을 확인하였으므로, 계절의 날씨 특성에 따른 화재발생 여부 분석을 이어가도록 하였음



3.4 온습도와 화재발생과의 상관계수 시각화

4) 종관기상관측 데이터와 화재발생 데이터의 상관계수는 아래와 같음

- 흰색은 범주형 데이터 및 텍스트가 포함된 데이터로 상관계수 분석이 불가능한 데이터임
- 색이 진해질수록 음의 상관계수를 지녔으며, 색이 연해질수록 양의 상관계수를 지닌 것으로 확인하였음
- 아래 그림에서 미루어 보았을 때, 종관기상관측 데이터와 화재발생 데이터의 상관관계는 유의미하지 않은 것으로 파악되었으나, 로지스틱 회귀분석을 통하여 분석을 이어 나가기로 하였음



3.5 종관기상관측 데이터를 바탕으로 화재발생 예측

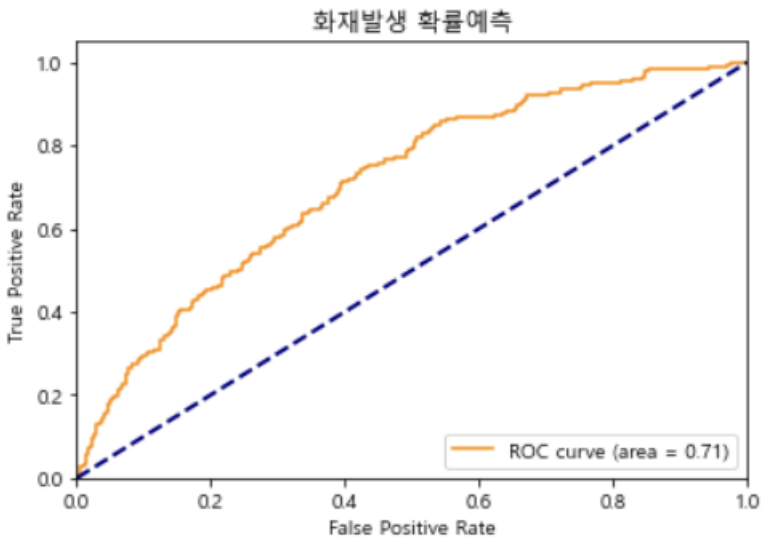
3) 종관기상관측 데이터로 로지스틱 회귀분석을 시행한 결과는 아래와 같음

- 로지스틱 회귀분석은 scikit-learn의 train_test_split을 사용하여 학습데이터와 테스트데이터의 비율을 7:3으로 지정하였음

- 아래의 오분류표를 보았을 때, 위음성은 0건이 나와서 모델이 적합하게 동작하는 듯하였지만, 진음성 역시 0건으로 나와 모델이 정상적으로 작동하지 않는 것을 확인하였음

- ROC 곡선을 그려 분류모형의 예측정확도를 평가하면, 어느정도 정상적으로 판단하는 것을 확인하였음

```
confusion_matrix
 [[10342     0]
 [  181     0]]
accuracy_score
 0.9827995818682885
precision_score
 0.0
recall_score
 0.0
f1_score
 0.9827995818682885
```



4. 요약 및 결론

- 화재발생 데이터와 종관기상관측 데이터를 이용하여 자료분석 및 시각화와 기계학습을 하고자 했는데, 화재발생 당시 발화요인은 날씨에 의한 화재 여부에는 큰 영향을 미치지 못함
- 4계절 중 봄, 겨울이 화재 건수가 특히 높은 것을 확인하였으므로, 계절의 날씨 특성에 따른 화재발생 여부 분석을 이어가도록 하였음
- 종관기상관측 데이터와 화재발생 데이터의 상관관계는 유의미하지 않은 것으로 파악됨
- 로지스틱 회귀분석으로 기계학습을 진행하였을 때, 분류 모형의 예측 정확도를 평가하면 어느정도 정상적으로 판단하는 것을 확인함
- 따라서, 날씨와 화재발생 데이터를 사용하여 화재발생을 예측한 결과, 모델의 적합성이 일정부분 있는 것으로 보임
- 향후 날씨 데이터 이외의 데이터를 바탕으로 화재 데이터와 상관계수 분석을 하여 모델을 만들면 모델의 적합성이 높아질 것으로 보임

참고문헌

1. 김진석 (2022) – 인공지능 강의노트
2. 공공데이터포털 소방청 화재발생 주소정보 파일데이터, https://www.data.go.kr/data/15044005/fileData.do
3. 기상청 기상자료개방포털 종관기상관측(ASOS) 파일셋, https://data.kma.go.kr/data/grnd/selectAsosRltmList.do?pgmNo=36&tabNo=1
4. matplotlib 파이 차트 그리기, https://hleecaster.com/python-matplotlib-pie-chart/
5. python) treemap 알아보기, https://data-newbie.tistory.com/731

부록

1

```
from sklearn.metrics import confusion_matrix, ₩
    accuracy_score, f1_score, precision_score, recall_score, ₩
    roc_auc_score, roc_curve, auc

print("confusion_matrix₩n", confusion_matrix(y_te, y_pred))
print("accuracy_score₩n",accuracy_score(y_te, y_pred))
print("precision_score₩n",precision_score(y_te, y_pred))
print("recall_score₩n",recall_score(y_te, y_pred))
print("f1_score₩n",accuracy_score(y_te, y_pred))
```

```
confusion_matrix
 [[10342     0]
 [  181     0]]
accuracy_score
 0.9827995818682885
precision_score
 0.0
recall_score
 0.0
f1_score
 0.9827995818682885
```

```
y_pred2 = clf.predict_proba(X_te)
fpr, tpr, _ = roc_curve(y_te, y_pred2[:, 1])
roc_auc = auc(fpr, tpr)
import matplotlib.pyplot as plt
plt.figure()
plt.plot(fpr, tpr,
    color="darkorange",
    label="ROC curve (area = %0.2f)" % roc_auc,
)
plt.plot([0, 1], [0, 1], color="navy", lw=2, linestyle="--")
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("화재발생 확률예측")
plt.legend(loc="lower right")
plt.show()
```