



머신러닝을 이용한 마약 범죄 통계 분석

빅데이터분석학 정희수



목차

- 01 연구 의의
- 02 데이터 소개
- 03 연구 과정 및 결과
- 04 기대효과
- 05 참고문헌

01 연구 의의

1. 마약 범죄의 증가
2. 마약 예방 강사 및 예방 제도 부족
3. 특히 “청소년” 범죄 위험 수위



01 연구 의의

[건전한 사회, 행복한 가정] 일상 파고든 마약... "전문적 교육 필요"

사회 > 사건·사고

투약 뿐 아니라 '운반책'에 뛰어드는 10대들...막을 방법은[김
동규의 마약 스톱!]

24시간 마약 중독 전화 상담 '1342', 운영 2개월 만에 900
여건

중독자 재활, 오남용 예방 등 약 900여건 상담 진행

(서울=뉴스1) 강승지 기자 | 2024-05-22 09:55 송고

댓글

가



02 데이터


1. 경찰청 홈페이지에서 수집
2. 데이터는 “마약범죄” 유형 기준으로 새로운 DB 생성. 12년치 데이터 수집.
3. 범죄 발생 지역, 발생 장소, 범행동기, 발생 시간대 등 총 10개 카테고리 사용

02 데이터


1. 경찰청 범죄통계

▶ 범죄발생 상황 관련 특성


1. 범죄 발생시간 및 장소

PDF 다운로드 

2. 범행 수법 및 도구


PDF 다운로드 

3. 피해결과


PDF 다운로드 

▶ 범죄자 특성


1. 사회인구학적 특성

PDF 다운로드 

2. 전과관련 특성


PDF 다운로드 

3. 기타


PDF 다운로드 

▶ 피해자 특성


1. 피해자 성별 연령

PDF 다운로드 

2. 피해자 피해시 상황

PDF 다운로드 

3. 범죄자와 피해자와의 관계

PDF 다운로드 

02 데이터

발생지

경기	경기	경기	전라	경기	충청	충청	경상	경기	경상	경기
부천	수원	성남	전주	안양	청주	마산	창원	광명	포항	안산

```
time= sheet_dataframes["시간"]
print(time)
```

	시간	2022	2021	2020	2019	2018	2017	2016	2015	2014	2013
0	0-3시	353	318	333	244	263	307	244	178	121	148
1	3-6시	303	246	214	214	155	202	184	135	97	106
2	6-9시	218	188	176	151	123	148	145	92	44	66
3	9-12시	806	645	753	635	529	671	591	476	312	325
4	12-15시	1035	758	941	773	724	743	641	578	357	320
5	15-18시	1155	808	1096	890	789	853	837	572	484	460
6	18-21시	1234	932	832	776	658	843	829	611	548	511
7	21-24시	1296	858	830	674	639	769	732	548	453	466

2011
0 170
1 85
2 43
3 259
4 215
5 340
6 352
7 431

```
region= sheet_dataframes["지역"]
print(region)
```

	지역	2022	2021	2020	2019	2018	2017	2016	2015	2014	2013	2012	₩
0	서울	2821	2275	2154	1884	1544	1746	1449	1360	1098	877	698	
1	부산	856	585	865	716	768	971	963	759	663	669	777	
2	대구	306	258	285	295	266	258	334	393	324	343	213	
3	인천	1062	610	759	709	558	586	641	557	389	356	495	
4	광주	129	159	126	76	56	31	75	69	26	45	40	
5	대전	101	96	113	118	105	97	117	78	30	95	45	
6	울산	107	68	73	79	52	73	78	67	81	41	48	
7	세종	29	40	34	31	21	19	8	2	0	0	0	
8	경기도	1952	1844	1930	1781	1222	1430	1255	1399	709	626	554	
9	강원도	194	100	159	172	117	140	236	110	56	60	49	
10	충청도	538	392	477	392	351	436	436	299	165	85	88	
11	전라도	225	203	230	159	109	122	165	105	91	76	59	
12	경상도	889	792	893	799	692	856	872	662	428	470	470	
13	제주도	51	35	28	26	18	38	39	26	29	22	27	
14	기타도시	71	70	135	51	17	25	19	16	355	403	379	
15	도시이외	1000	564	925	750	620	628	720	509	362	430	323	

2011
0 716
1 702
2 262
3 296
4 51
5 63
6 43
7 0

03

연구진행 및 결과

1. 데이터 전처리 및 새로운 DB 생성
2. Train: Test 비율 정하기
3. 랜덤포레스트 모델 사용
4. 모델 성능 평가



03 연구 과정 및 결과

```
[13] # 데이터의 분포를 기반으로 가상의 데이터를 생성하는 함수
def generate_fake_data(real_data, num_samples):
    fake_data = real_data.copy()
    for col in real_data.columns:
        fake_data[col] = np.random.choice(real_data[col], num_samples)
    return fake_data
```

```
[14] # 가상의 비범죄 데이터 생성
fake_dataframes = {}
for sheet_name, df in sheet_dataframes.items():
    num_fake_samples = len(df) # 실제 데이터와
    fake_df = generate_fake_data(df, num_fake_samples)
    fake_df['범죄 발생 여부'] = 0
    fake_dataframes[sheet_name] = fake_df
```

```
# 실제 데이터에도 라벨 추가
for sheet_name, df in sheet_dataframes.items():
    df['범죄 발생 여부'] = 1
```

1. 원데이터에 기반하여 가상 비범죄 데이터 생성
2. 원데이터와 가상 비범죄 데이터 라벨링
3. 하나의 데이터로 통합

```
[15] # 실제 데이터와 가상의 데이터를 결합
combined_dataframes = {}
for sheet_name in sheet_dataframes.keys():
    combined_df = pd.concat([sheet_dataframes[sheet_name], fake_dataframes[sheet_name]], ignore_index=True)
    combined_df.columns = combined_df.columns.astype(str)
    combined_dataframes[sheet_name] = combined_df
```

```
[16] # 인코딩 및 전처리
label_encoders = {}
for sheet_name, df in combined_dataframes.items():
    le_dict = {}
    for column in df.select_dtypes(include=['object']).columns:
        if column != '범죄 발생 여부':
            df[column] = df[column].astype(str)
            le = LabelEncoder()
            df[column] = le.fit_transform(df[column])
            le_dict[column] = le
    label_encoders[sheet_name] = le_dict
```

03 연구 과정 및 결과

```
[17] # 피처와 라벨 분리 및 학습/테스트 데이터 분할
```

```
X_train_list = []  
X_test_list = []  
y_train_list = []  
y_test_list = []  
feature_names = {}
```

```
for sheet_name, df in combined_dataframes.items():
```

```
    X = df.drop(columns=['범죄 발생 여부'])
```

```
    y = df['범죄 발생 여부']
```

```
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

```
    X_train_list.append(X_train)
```

```
    X_test_list.append(X_test)
```

```
    y_train_list.append(y_train)
```

```
    y_test_list.append(y_test)
```

```
    feature_names[sheet_name] = X.columns.tolist()
```

```
[18] # 랜덤포레스트 모델 학습
```

```
models = []
```

```
for i in range(len(X_train_list)):
```

```
    model = RandomForestClassifier(n_estimators=100, random_state=42)
```

```
    model.fit(X_train_list[i], y_train_list[i])
```

```
    models.append(model)
```

```
[19] # 모델 평가
```

```
for i in range(len(models)):
```

```
    y_pred = models[i].predict(X_test_list[i])
```

```
    print(f"Model for sheet {list(combined_dataframes.keys())[i]}")
```

```
    print(f"Accuracy: {accuracy_score(y_test_list[i], y_pred)}")
```

```
    print(f"Confusion Matrix:\n {confusion_matrix(y_test_list[i], y_pred)}")
```

```
    print(f"Classification Report:\n {classification_report(y_test_list[i], y_pred)}")
```

4. 데이터 train: test = 7:3 분할

5. 모델 학습

6. 모델 평가 진행

03 연구 과정 및 결과

```
Model for sheet 교육
Accuracy: 0.6666666666666666
Confusion Matrix:
[[4 1]
 [3 4]]
Classification Report:
              precision    recall  f1-score
0           0.57         0.80     0.67
1           0.80         0.57     0.67
```

```
Model for sheet 국적
Accuracy: 0.7272727272727273
Confusion Matrix:
[[3 2]
 [1 5]]
Classification Report:
              precision    recall  f1-score
0           0.75         0.60     0.67
1           0.71         0.83     0.77
```

데이터의 양이 충분하며 분류도 다양하다면
정확도가 어느 정도 높은 것으로 보임

```
Model for sheet 장소
Accuracy: 0.8571428571428571
Confusion Matrix:
[[ 8  3]
 [ 0 10]]
Classification Report:
              precision    recall  f1-score
0           1.00         0.73     0.84
1           0.77         1.00     0.87
```

03 연구 과정 및 결과

Model for sheet 요일

Accuracy: 0.2

Confusion Matrix:

[[0 3]

[1 1]]

Classification Report:

	precision	recall	f1-score
--	-----------	--------	----------

0	0.00	0.00	0.00
---	------	------	------

1	0.25	0.50	0.33
---	------	------	------

Model for sheet 자백

Accuracy: 0.3333333333333333

Confusion Matrix:

[[1 0]

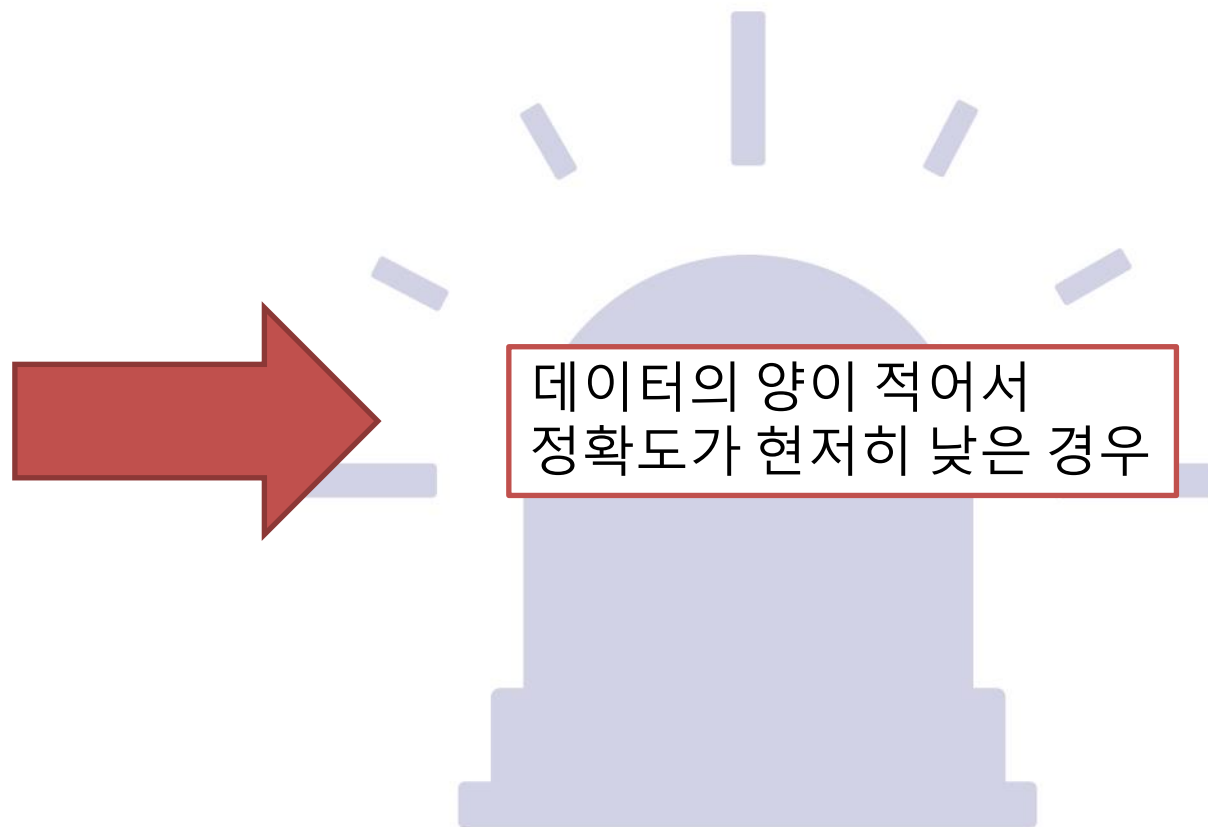
[2 0]]

Classification Report:

	precision	recall	f1-score
--	-----------	--------	----------

0	0.33	1.00	0.50
---	------	------	------

1	0.00	0.00	0.00
---	------	------	------



03 연구 과정 및 결과

```
[20] new_data = pd.DataFrame({
    '시간': [label_encoders['시간']['시간'].transform(['15-18시'])[0]],
    '요일': [label_encoders['요일']['요일'].transform(['월'])[0]],
    '장소': [label_encoders['장소']['장소'].transform(['단독주택'])[0]],
    '지역': [label_encoders['지역']['지역'].transform(['서울'])[0]],
    '연령': [label_encoders['연령']['연령'].transform(['18'])[0]],
    '부모': [label_encoders['부모']['부모'].transform(['실양부모'])[0]],
    '교육': [label_encoders['교육']['교육'].transform(['대학졸업'])[0]],
    '동기': [label_encoders['동기']['동기'].transform(['유혹'])[0]],
    '자백': [label_encoders['자백']['자백'].transform(['일부자백'])[0]]
})
new_data.columns = new_data.columns.astype(str) # 새로운 데이터의 열 이름을 문자열로 변환
```

<예측 예시1>

```
[21] # 예측
for i in range(len(models)):
    # 새로운 데이터의 피쳐 이름을 학습된 피쳐 이름과 일치시킴
    X_new = new_data.reindex(columns=feature_names[list(combined_dataframes.keys())[i]], fill_value=0)
    prediction = models[i].predict(X_new)
    sheet_name = list(combined_dataframes.keys())[i]
    print(f"Prediction for sheet {sheet_name}: {'범죄 발생' if prediction[0] == 1 else '범죄 미발생'})
```

➡ Prediction for sheet 시간: 범죄 미발생
Prediction for sheet 요일: 범죄 미발생
Prediction for sheet 지역: 범죄 발생
Prediction for sheet 장소: 범죄 발생
Prediction for sheet 연령: 범죄 발생
Prediction for sheet 부모: 범죄 발생
Prediction for sheet 교육: 범죄 발생
Prediction for sheet 국적: 범죄 발생
Prediction for sheet 동기: 범죄 발생
Prediction for sheet 자백: 범죄 발생

03 연구 과정 및 결과

```
[95] new_data = pd.DataFrame({
    '시간': [label_encoders['시간']['시간'].transform(['15-18시'])[0]],
    '요일': [label_encoders['요일']['요일'].transform(['월'])[0]],
    '장소': [label_encoders['장소']['장소'].transform(['단독주택'])[0]],
    '지역': [label_encoders['지역']['지역'].transform(['서울'])[0]],
    '연령': [label_encoders['연령']['연령'].transform(['18'])[0]],
    '부모': [label_encoders['부모']['부모'].transform(['실양부모'])[0]],
    '교육': [label_encoders['교육']['교육'].transform(['중등졸업'])[0]],
    '동기': [label_encoders['동기']['동기'].transform(['유혹'])[0]],
    '국적': [label_encoders['국적']['국적'].transform(['일본'])[0]],
    '자백': [label_encoders['자백']['자백'].transform(['일부자백'])[0]]
})
new_data.columns = new_data.columns.astype(str) # 새로운 데이터의 열 이름을 문자열로 변환
```

<예측 예시2>

```
[96] # 예측
for i in range(len(models)):
    # 새로운 데이터의 피쳐 이름을 학습된 피쳐 이름과 일치시킴
    X_new = new_data.reindex(columns=feature_names[list(combined_dataframes.keys())[i]], fill_value=0)
    prediction = models[i].predict(X_new)
    sheet_name = list(combined_dataframes.keys())[i]
    print(f"Prediction for sheet {sheet_name}: {'범죄 발생' if prediction[0] == 1 else '범죄 미발생'})
```

☞ Prediction for sheet 시간: 범죄 발생
Prediction for sheet 요일: 범죄 미발생
Prediction for sheet 지역: 범죄 발생
Prediction for sheet 장소: 범죄 발생
Prediction for sheet 연령: 범죄 미발생
Prediction for sheet 부모: 범죄 발생
Prediction for sheet 교육: 범죄 발생
Prediction for sheet 국적: 범죄 발생
Prediction for sheet 동기: 범죄 발생
Prediction for sheet 자백: 범죄 발생

04 기대효과

1. 데이터의 다양화와 규모화
2. 식품의약품 데이터 활용
3. 인공지능 활용



04 기대효과

1. 현재 데이터는 선택된 카테고리만 원하는 데이터만 추출하여 만든 DB → 더 많은 카테고리 데이터의 양이 방대해진다면 모델 학습 시 정확도가 높아질 것으로 예상
2. 마약범죄의 원인 중 의료용 마약 처방도 큰 비중을 차지 → 식의약처 DB 사용 → 의료용 마약 쇼핑 방지 방안 마련
3. 24시간 마약 중독 전화 상담 데이터 분석 → 실질적인 대안 방안 강구에 도움
4. 인공지능 기술 활용 → 마약 사범 검거 및 추적 기술 개발
5. 소지 및 운반에 대한 예방 대책도 마련 가능

05

참고문헌



05 참고문헌

- [1] 김민선 “토픽 모델링을 활용한 코로나19 이후 청소년 마약 관련 이슈 분석”, 2023
- [2] 김시원, 이은효, 서대현, 김원빈, 주웅진 “서울시 스마트치안을 위한 머신러닝 기반 범죄예측 분석방안”
- [3] 정용찬 “범죄 데이터를 활용한 재범 예측 통계 기법 비교 분석(로지스틱 회귀분석, 랜덤포레스트를 중심으로), 2021
- [4] 강현우 “멀티모달 데이터 융합을 사용한 딥러닝 기반의 범죄 발생 예측”, 2017
- [5] 황윤재 “머신러닝을 사용한 촉발범죄 예측 및 분석(주거침입절도범죄를 중심으로), 2023
- [6] 허선영, 김주영, 문태현 “머신러닝 기반 범죄발생 위험지역 예측”, 2018
- [7] 이주원 “머신러닝 기반 야간에 발생한 범죄예측모형(환경요소의 상관관계를 중심으로), 2022
- [8] 공예은, 조유정, 최성철 “BigData 기반 범죄예측의 분석기법 연구”, 2020
- [9] <https://www.fnnews.com/news/202405171131471415>
- [10] <https://www.news1.kr/articles/5422918>
- [11] <https://www.joongboo.com/news/articleView.html?idxno=363653569>
- [12] <http://www.hitnews.co.kr/news/articleView.html?idxno=48921>
- [13] <https://www.etnews.com/20240503000192>



THANK YOU