Rangga Aulia Rahman
20/456849/TK/50673

Tugas Rekayasa Data 2

I. Deskripsi Dataset

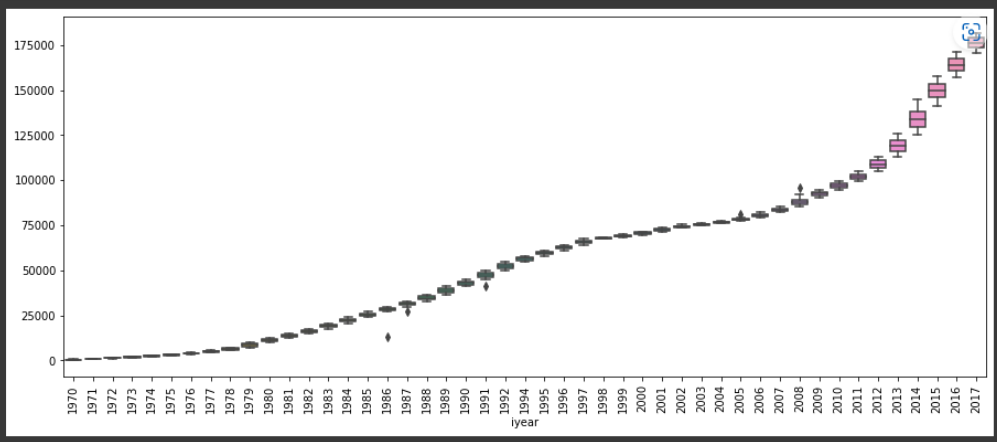
Dataset ini memiliki informasi terkait serangan teroris dari tahun 1970 hingga 2016. Terdapat informasi semisal mengenai tanggal terjadinya serangan, lokasi negara, anggota kelompok teror mana yang menyerang, hingga persenjataan dan juga korban dalam serangan tersebut.

II. Isi

Data Cleaning

Untuk proses data cleaning disini kita akan mencoba untuk mencari outlier dalam datanya. Disini kita berfokus ke salah satu atribut yakni jumlah korban pada serangan yang terjadi.

```
sns.boxplot(data=df, y=df['nkill'].index, x = 'iyear')
plt.show()
```



Ada beberapa tahun dengan outlier, tapi sekarang kita akan lihat ke 1986. Karena nampaknya outliernya juga lebih jauh dibandingkan tahun lain.

```
df_1986_sorted = df[(df['iyear'] == 1986)].sort_values(by='nkill', ascending=False)
df_1986_sorted['nkill']
```

```
27086    240.0
28561    227.0
27413     75.0
29812     62.0
27589     46.0
28200     45.0
27430     44.0
28579     41.0
29822     40.0
```

Disini memang terbukti, bahwa di 1986 ada jumlah yang outlier, jumlah lain berkisar di puluhan, namun ada 2 row dimana jumlah korban mencapai 200 korban. Sedikit hal untuk kritik ke diri saya sendiri, sepertinya boxplot tidak akurat, karena satuannya ribuan, ini kemungkinan disebabkan kesalahan saya dalam input fungsi untuk boxplot sns. Tapi terlepas dari kesalahan tersebut memang ada outlier di 1986.

Menghilangkan kedua outlier tersebut dilakukan dengan,

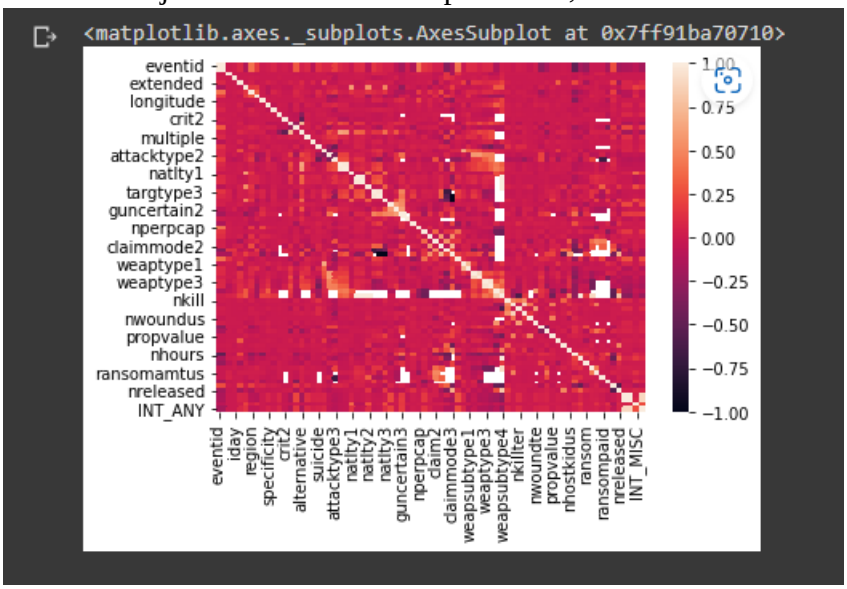
```
df = df.drop(df[(df['eventid'] == 198601130005) & (df['eventid'] == 198607010012)].index)
```

Rangga Aulia Rahman
20/456849/TK/50673

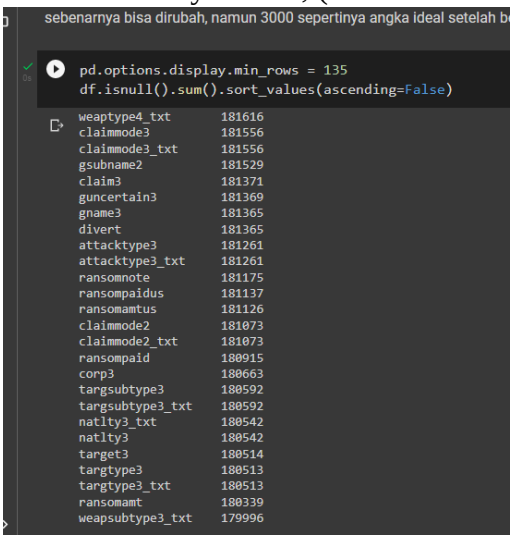
Namun ketika saya cek kembali outlier dengan jumlah 227 tetap muncul, dan jikapun di drop 1 1 tetap tidak bisa hilang. Sehingga kita hanya bisa menghapus 1 outlier dari temuan awal. Selanjutnya kita akan melihat atribut dalam dataset.

Data Integration

Jika kita cari atribut redundant, salah satunya bisa kita lihat dengan jumlah null atau N/A. Disini jika kita amati heatmap korelasi,



Kita bisa lihat contohnya atribut **nkill** memiliki korelasi kuat dengan beberapa atribut lain. Namun apakah korelasi ini memiliki **alasan** yang valid? Nyatanya sepertinya tidak. Hal ini karena dalam dataset ini ternyata sangat banyak nilai null di masing-masing column (lebih dari ribuan). Hal ini mengindikasikan jika, korelasi kuat ini terjadi karena nilai yang **sama-sama null** untuk kedua atribut yang berbeda di posisi yang sama. Misalnya, nkill memiliki puluhan ribu null, tapi weapsubtye4 juga memiliki ribuan null, meskipun tidak ada hubungannya, tapi heatmap menganggap ini berkorelasi karena ketika **nkill** null, **weapsubtype4** juga sama-sama null, sehingga membuat **ilusi** korelasi yang kuat. Perlu diingat juga bahwa korelasi *tidak selalu* membuat adanya sebab, (corelation does not equal causation).



Bisa terlihat bahwa sejatinya sangat banyak atribut yang redundant karena memuat sangat banyak null (dataset ini row nya sekitar 18.000, jadi hampir kosong semua di

setiap row). Hal ini mengindikasikan, bahwa sebenarnya untuk column dengan jumlah null yang sangat banyak tersebut, kontribusinya tidaklah terlalu signifikan, dan justru malah membuat kita *misled* untuk menganggap adanya korelasi antar atribut yang kuat. Sekarang kita akan melakukan proses untuk menghilangkan atribut redundant tersebut.

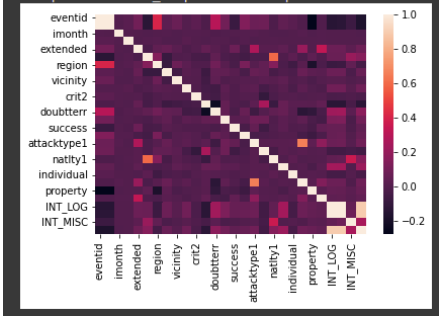
Data Reduction

```
[100] df_new = df.drop(df.columns[df.apply(lambda col: col.isnull().sum() > 3000)], axis=1)
      df_new
      #Ini artinya kita akan drop columns, dimana df.columns menjadi tujuannya, dan kita hanya akan tertarik
      #ke columns tertentu, yang dijalankan dengan .apply, ke Axis = 1 (axis column)
```

Kita akan hilangkan column dengan jumlah null diatas 3000. Dari jumlah column awal 135 kita akan menyisakan sekitar 39 column. Heatmap juga memberikan kita hasil bahwa korelasi antar data memang tidak begitu kuat. Disini namun juga kemungkinan ada trade off tergantung jumlah batas null yang kita inginkan, beberapa informasi penting mungkin justru terhapus.

```
correlation_new = df_new.corr()
sns.heatmap(correlation_new)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff912be20
```



III. Kesimpulan

a. Dataset ini mempunyai sangat banyak atribut redundant yang tidak begitu signifikan, sehingga hendaknya ditambahkan informasi mengenai *importance* dari adanya atribut tersebut.
b. Atribut redundant diatas menyebabkan adanya ilusi korelasi kuat antar atribut ketika sebenarnya tidak ada.
c. Utility dataset ini kemungkinan besar adalah untuk sekedar memberikan infografis mengenai serangan terror yang terjadi per tahun, di negara, atau berdasarkan aktor yang menyerang, karena korelasi antar data yang kurang kuat.

Link Colab

https://colab.research.google.com/drive/1Myoov_oGey_8I6iKE4EXBtwkEO9imF4n?usp=sharing