USA Gun Incidents

Data Mining Project Presentation, Group 10

Giacomo Aru
g.aru@studenti.unipi.it
Roll number: 597700

Giulia Ghisolfi
g.ghisolfi@studenti.unipi.it
Roll number: 664222

Luca Marini
l.marini11@studenti.unipi.it
Roll number: 578543

Irene Testa
i.testa@studenti.unipi.it
Roll number: 582061

Master Degree in Computer in Science, University of Pisa
Data mining course (309AA), Prof. Anna Monreale,
Academic Year: 2023-24

Outline

Data Understanding and Preparation

Clustering Analysis

Predictive Analysis

Explainability

Time Series Analysis

Data Understanding and Preparation

Exploratory analysis

We identified the following issues:

- **Duplicates** (there are sets of attributes that should uniquely identify each incident, e.g., `latitude`, `longitude`, `date`, `incidents_characteristics1` and `incidents_characteristics2`)
  $\Rightarrow$ *we dropped them*
- 78.31% of the records had at least an attribute with a **NaN** value
- Syntactic and semantic **inconsistencies**

Date Attribute (1)

- 9.6% records had **out of range** values (e.g. year equal to 2028)
- We explored different approaches to correct data but all all of them *significantly altered the distribution*
  $\Rightarrow$ we divided the attribute `date` into `day`, `month`, and `year`
  $\Rightarrow$ values out of range for the attribute `year` were set to NaN

Date Attribute (2)

After **data cleaning** we could observe that:

- months with *lower* number of incidents: April, June and November
- months with *higher* number of incidents: January, March and August
- days of the week with *lower* number of incidents: Thursday and Friday
- days of the week with *higher* number of incidents: Saturday and Sunday
- holidays with *lower* number of incidents: Christmas and Thanksgiving
- holidays with *higher* number of incidents: New year's eve and Independence day

Geospatial attributes (1)

Issues:

- some points located in **China**
- records with equal `latitude` and `longitude` but different `city_or_county`
- inconsistencies in the attribute `congressional_district` between *year_state_district_house.csv* and *incidents.csv* for states with a single congressional district
  $\Rightarrow$ we set it to **0** in *incidents.csv*
- inconsistencies in the count of congressional districts per state
  $\Rightarrow$ we set `congressional_district` to **NaN** in *incidents.csv*
- given a pair of values in the `latitude` and `longitude`, the `congressional_district` attribute did not consistently maintain the same value
  $\Rightarrow$ we set it to the most frequently occurring value
- inconsistencies in `state_house_district` and `state_senate_district`
  $\Rightarrow$ **left unaltered** since not used in further analysis

Geospatial attributes (2)

Correction of the attributes:

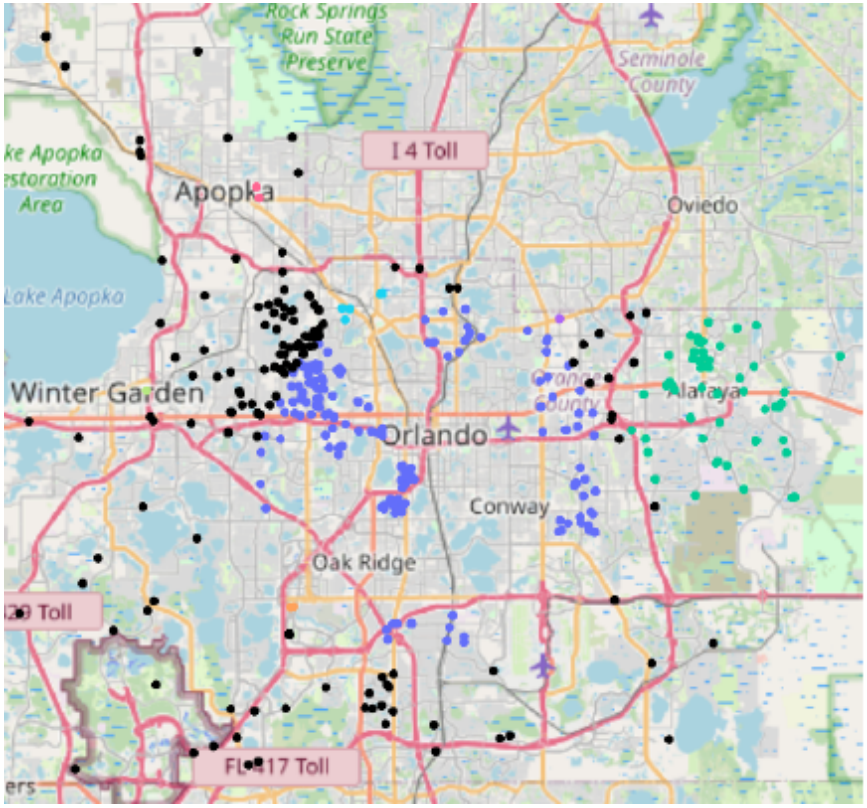
- We used **GeoPy** library to retrieve for each latitude and longitude:
  - `importance`
  - `address_type`
  - state (`state_geopy`)
  - county (`county_geopy`)
  - suburb (`suburb_geopy`)
  - city (`city_geopy`)
  - town (`town_geopy`)
  - village (`village_geopy`)
  - address (`display_name`)
- We downloaded a **list of Counties** from *Wikipedia*
  - we used it to check if the county belonged to the state when incident data didn't align with `Geopy` information or when `latitude` and `longitude` were not available

Geospatial attributes (3)

Inference of the attribute `city`:

1. We computed **city centroids**
2. For each point to infer we searched for the city with the **closest centroid**
3. If such distance fell within the 75^{th} percentile of the distances between all the points assigned to it, we set the corresponding city

Through this process, we inferred **2249** missing values.

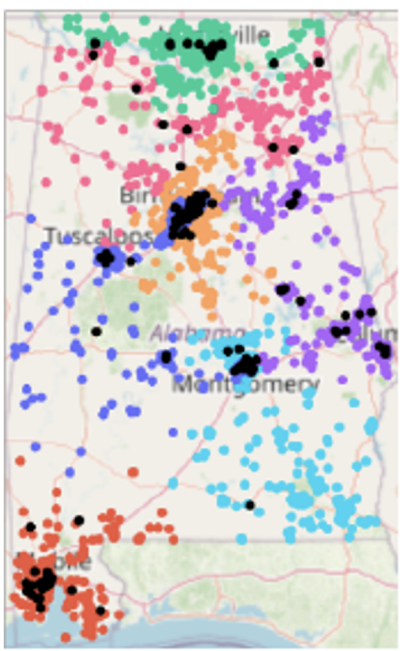


Incidents in Orange County in Florida. Colored dots represents incidents whose city was inferred. Black points denote those whose city was left as 'NaN' even after the inference process.

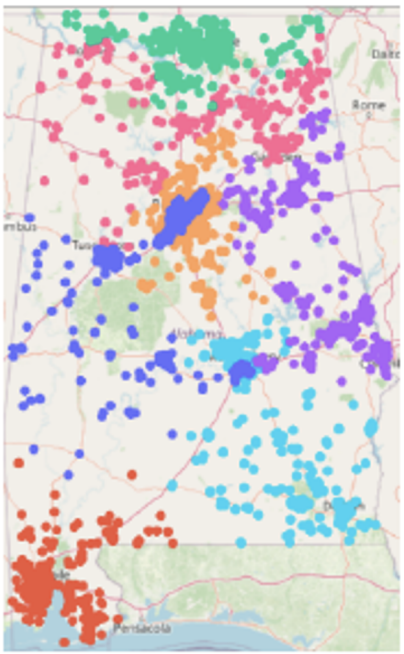
Geospatial attributes (4)

Inference of the attribute `congressional_district`:

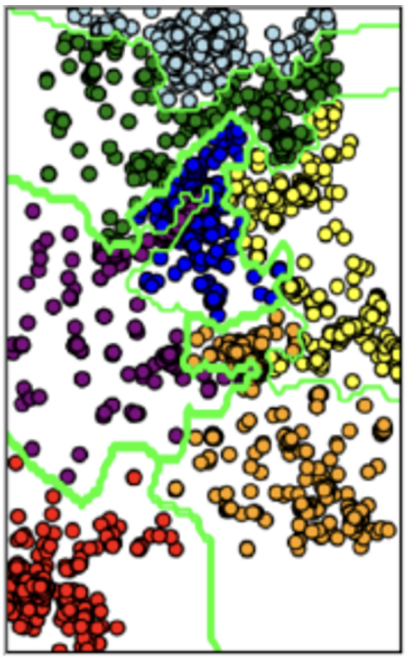
- We use **KNN** (*K=1* and *geodesic distance*) to infer missing `congressional_district` values
  - Every congressional district in a state had **at least one incident**
  - This attribute is essential for **joining** the 'incidents.csv' dataset with the 'year_state_district_house.csv' dataset, and given its definition it is **valuable for computing local indicators**
  - Through this process, we inferred **2893** missing values.



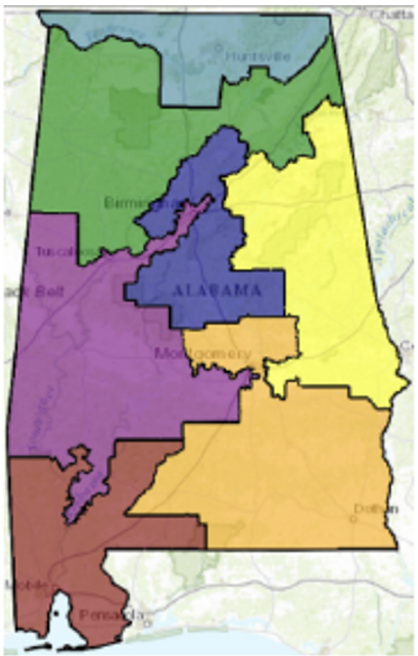
Incidents colored coded based on congressional district **before** the inference (black stands for NaN)



Incidents colored coded based on congressional district **after** the inference



KNN decision boundaries



Actual boundaries

Attributes about participants characteristics (1)

Issues:

- instances of **strings in numerical fields**
- **negative values** for counts or ages
- ages exceeding **150 years**
- minimum ages surpassing the maximum or average age
- inconsistencies between minimum, maximum, and average age of the participants and the number of children, teens, and adults
- number of participants inferior than number of individuals in a subcategory

Attributes about participants characteristics (2)

Cleaning and inference:

- at first we set **inconsistent values to NaN**
- in some cases total number of participants was inferred by **summing the number of male and female participants**
- when the data of the random participant was consistent and `n_participants` $= 1$, we set attributes related to **age** and **gender**

Attributes about incident characteristics (1)

From `incidents_characteristics1` and `incidents_characteristics2` we created the following **tags**:

- `firearm`
- `shots`
- `air_gun`
- `aggression`
- `defensive`
- `suicide`
- `unintentional`
- `abduction`
- `injuries`
- `death`
- `illegal_holding`
- `drugs`
- `officers`
- `organized`
- `social_reasons`
- `road`
- `house`
- `school`
- `children`
- `workplace`

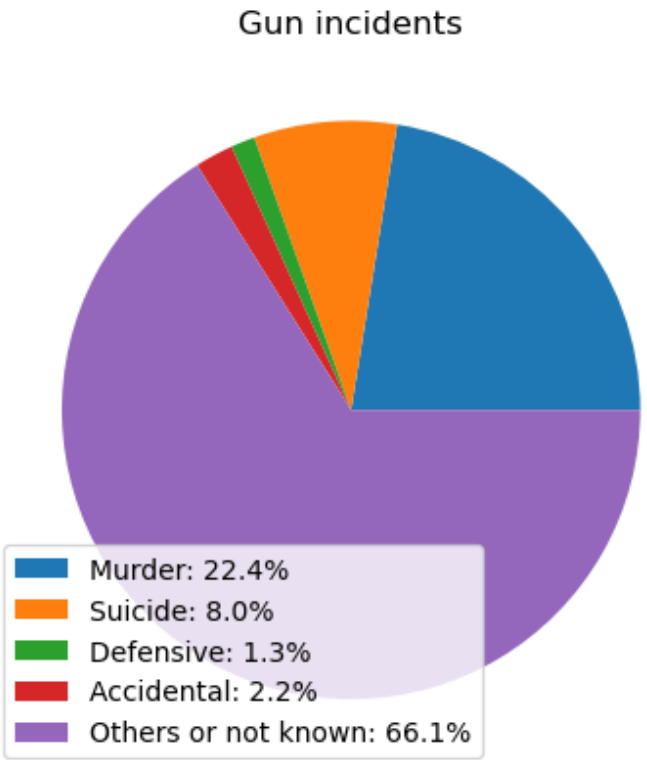
In the absence of sufficient information we set the corresponding variables to **False**.

Attributes about incident characteristics (2)

Statistics on incident characteristics:

For **fatal** incidents:

- 22.4% are *murders*
- 8% are *suicides*
- 1.3% involve *self-defense*
- 2.2% are *accidental*
- 66.1% belong to the category *others*



- incidents involving *women* exhibit a distribution similar to the one for incidents involving men, but with **most suicides** and **lower officers involvement**

Poverty dataset

- Wyoming had two entries for year 2009 and none for 2010
  $\Rightarrow$ since the dataset is ordered by year we set the second to 2010
- there was no data for the year 2012
  $\Rightarrow$ we calculated the average of the `povertyPercentage` values for the years 2011 and 2013
- New Hampshire had the **lower** average poverty percentage
- Mississippi had the **highest** average poverty percentage

Elections dataset

- District of Columbia only provided data for 2020
  $\Rightarrow$ we get missing values from **Wikipedia**
- number of `totalVotes` for 2020 differed from the ones in Wikipedia, while votes received by winning party didn't
  $\Rightarrow$ we set data for `totalVotes` using values in Wikipedia
- **upper outlier** in `candidateVotes` in Maine
  $\Rightarrow$ corrected using Wikipedia
- some congressional districts in Florida had values of **0**, **1** or **-1** for `candidateVotes` and `totalVotes` for 2014
  $\Rightarrow$ we copied values from previous year (according to external sources it means **no candidates filed** to challenge the representative)

Joint datasets analysis

- no more outliers or erroneous data
- negative correlation between `povertyPercentage` and `latitude`
- the number of incidents per 100k inhabitants in the District of Columbia is **significantly higher** than in other states (especially in January 2014)
- in 2013 the number of incidents is **low** in all states
- highest number of incidents per month happened in August 2014 in Delaware and in July 2015 in Wyoming
- *states* with **higher** number of incidents are Alaska, Delaware, Illinois, Louisiana, and South Carolina
- *states* with **lower** number of incidents are Hawaii, Idaho, Arizona, and California

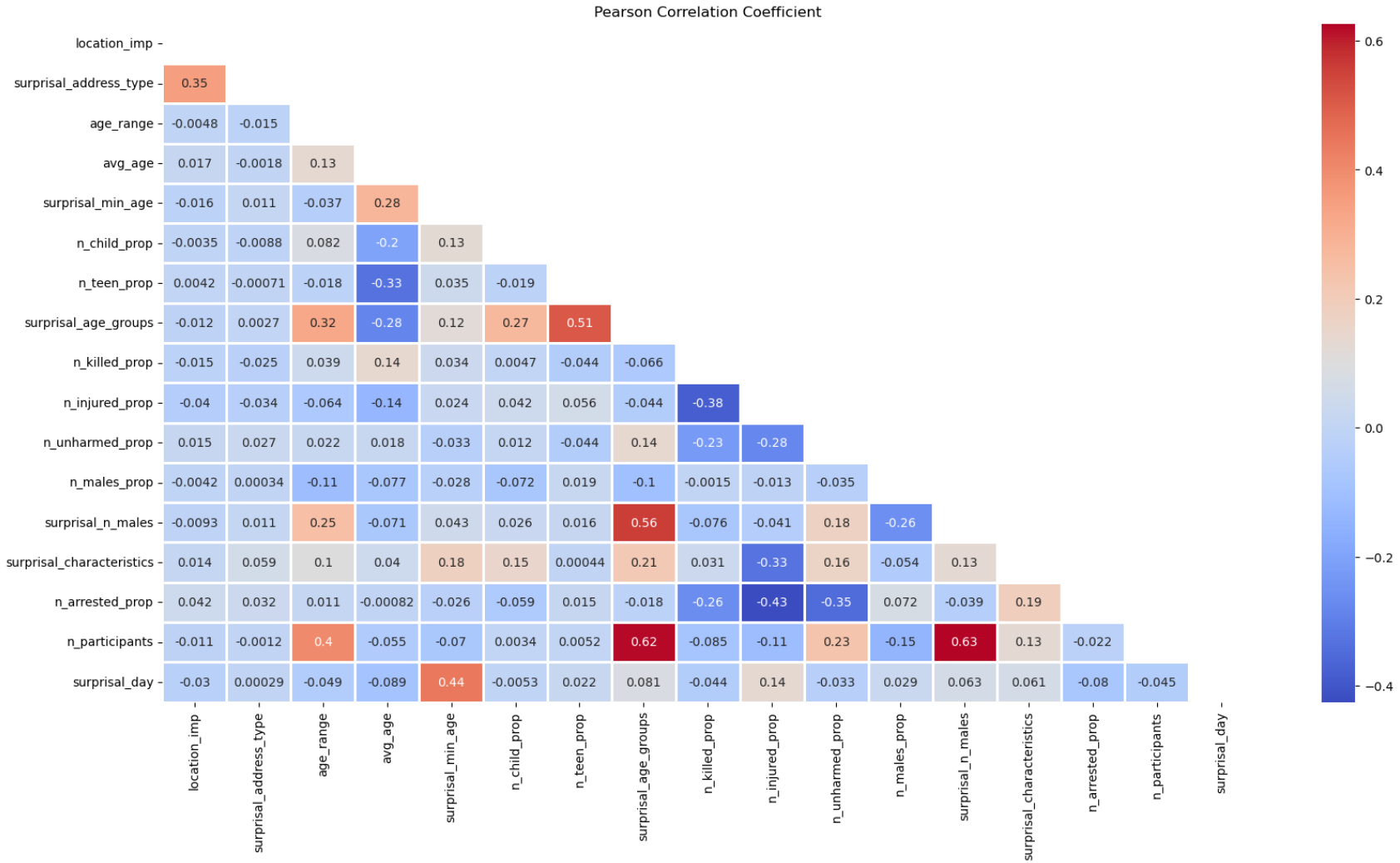
Clustering Analysis

Features for clustering (1)

We identified a significant set of **uncorrelated** dataset features and calculated additional **indicators**:

- `age_range` and `avg_age` to effectively represent the ages of the people involved in the incident
- **Ratios** between the number of people involved in the incident with a specific characteristic and the total number of participants (e.g. `n_killed_prop`, `n_injured_prop`, ...)
- Indicators based on the concept of **surprisal** for both numerical and categorical features as well as for sets of features (e.g. `surprisal_min_age`, `surprisal_characteristics`, ...)

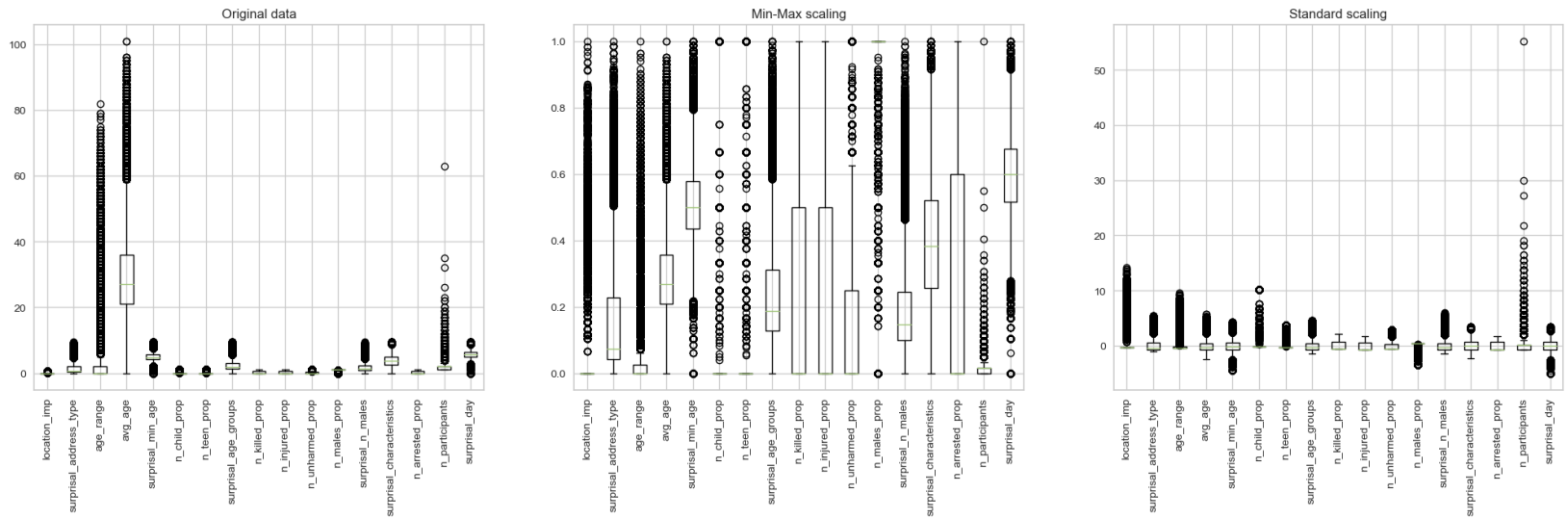
Features for clustering (2)



Pearson Correlation Coefficient between the features used for clustering. It ranges between [-0.43, 0.63].

Features for clustering (3)

- The number of records without NaN values on the computed indicators amounts to 131811
- 235 records are duplicated (share the same values on the indicators)
- Potential inconsistencies were addressed during the data preparation phase
- Outliers were not excluded because we believe that such instances contain valuable information
- We opted for Min-Max Scaling



Indicators distributions.

Clustering Algorithms

We applied several clustering algorithms to the dataset:

- K-Means, Bisecting K-Means, X-Means
- Agglomerative Hierarchical Clustering (with different linkage methods)
- DBSCAN
- Self Organizing Maps (SOM)

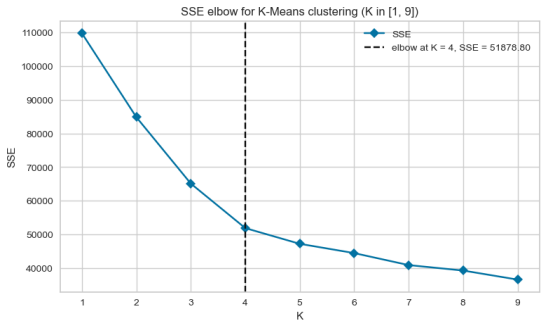
Hierarchical Clustering and DBSCAN were used to cluster only the incidents that happened in the state of **Illinois**:

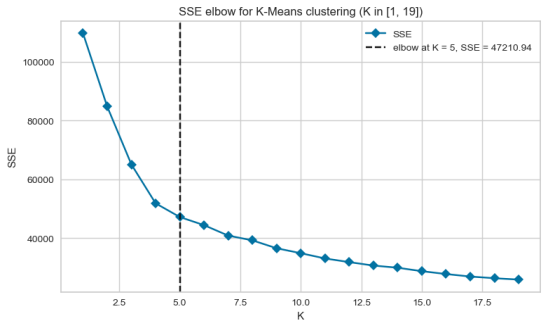
- it has fewer missing values
- the feature distribution closely resembles that of the entire dataset

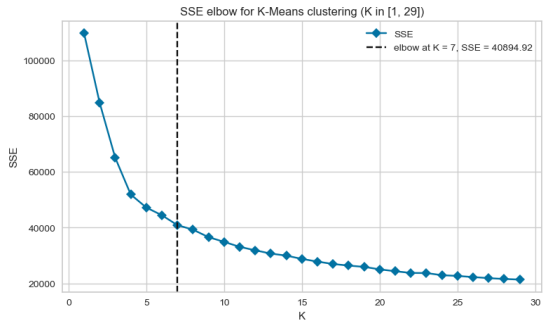
K-Means

Identification of the best value of k:

- plot of SSE (for K $\in$ [1,30]), Silhouette and Calinski-Harabasz score varying K
- X-Means with BIC and MDL scores (stop at K=30)
- evaluation of BSS and Davies-Bouldin score







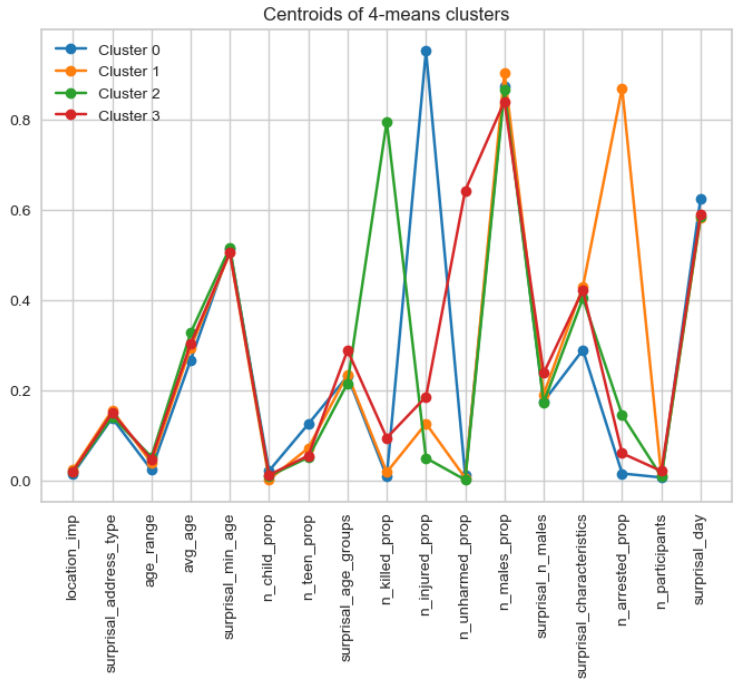
Most of the methods were concordant in identifying K = 4 as the optimal number of clusters.

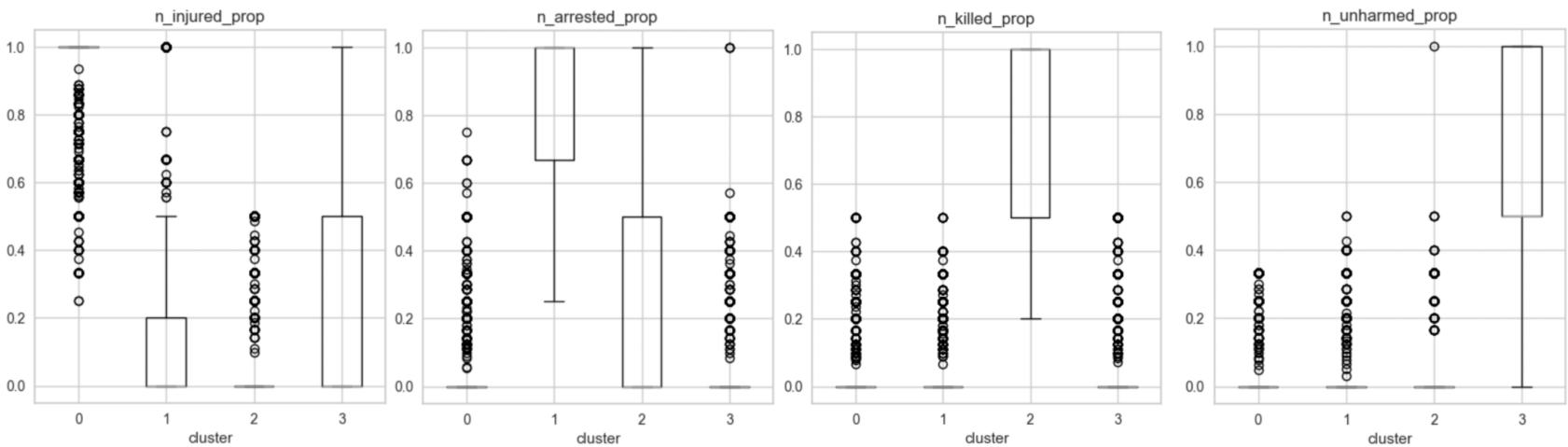
K	SSE	BSS	Calinski-Harabasz	Davies-Bouldin	Silhouette	# Iterations	Cluster sizes
4	5.188×10^4	5.786×10^4	4.900×10^4	1.213	0.327	6	1: 40711; 2: 31460; 3: 31389; 0: 28251
5	4.721×10^4	6.253×10^4	4.364×10^4	1.456	0.299	6	2: 30377; 1: 28640; 3: 27375;4: 25979; 0: 19440
7	4.089×10^4	6.885×10^4	3.698×10^4	1.397	0.303	21	3: 28513; 5: 26781; 6: 20842; 0: 20202; 1: 19382; 2: 10031; 4: 6060

K-Means (1)

Characterization of the clusters:

- The features that mostly differ in clusters are `n_injured_prop`, `n_arrested_prop`, `n_killed_prop` and `n_unharmed_prop`

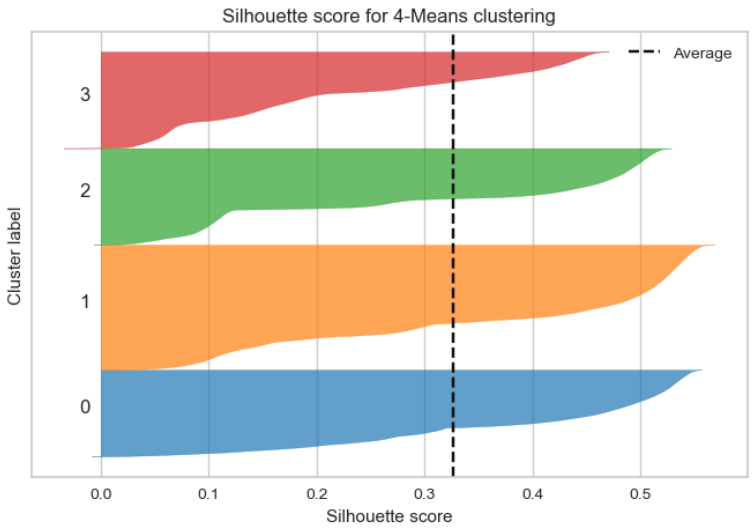


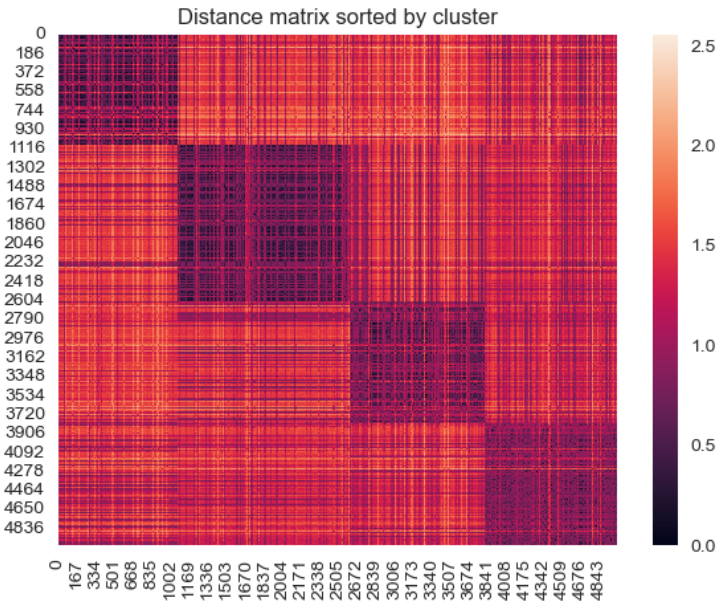


K-Means (2)

Evaluation of the clustering results:

- The average Silhouette score amounts to **0.327**
- The Pearson Correlation Coefficient between the distance matrix and the ideal distance matrix is equal to **0.62**
- To mitigate the effect of the initialization of the centroids we attempted initializing them with the final centroids computed by Bisecting K-Means, but this yielded comparable results

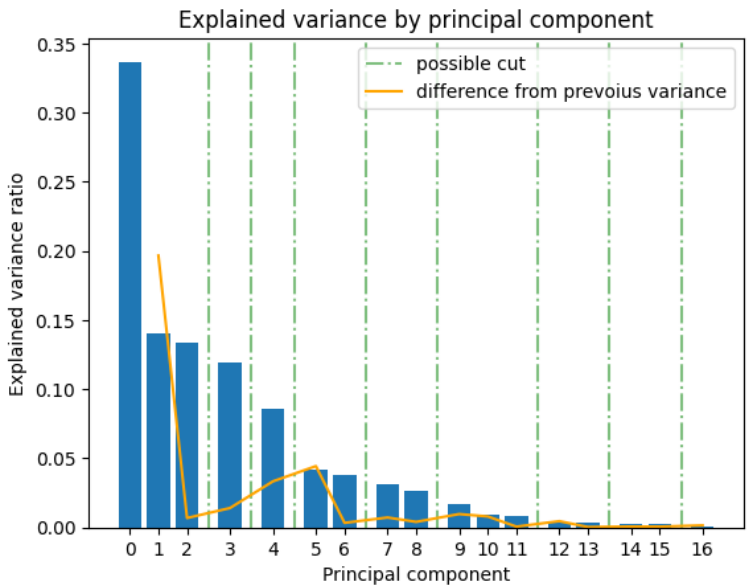




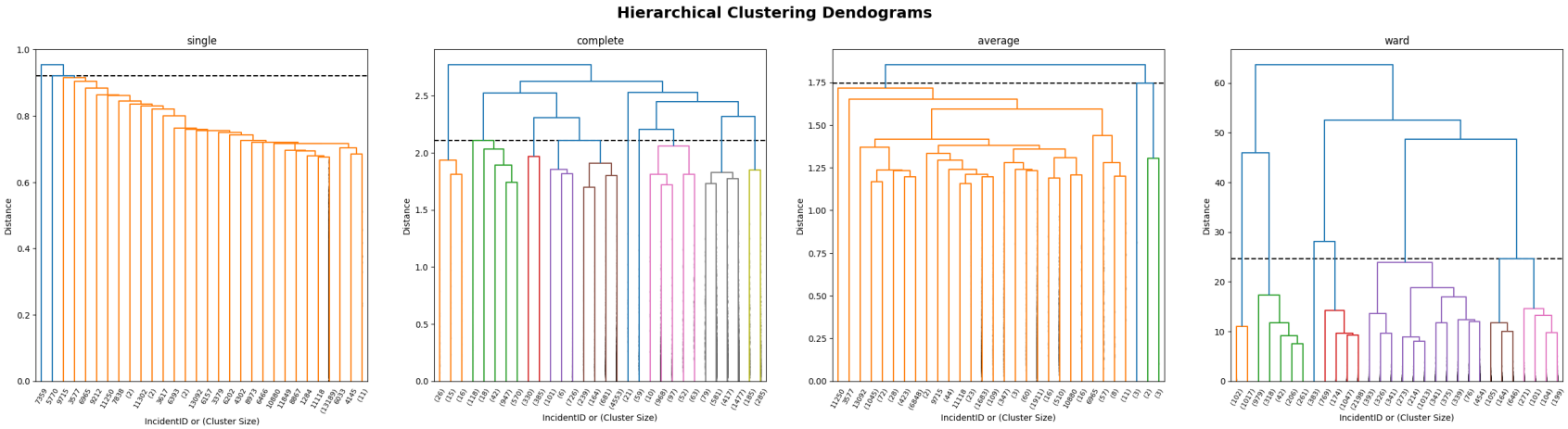
K-Means (3)

We also attempted to apply K-Means after transforming the data using Principal Component Analysis (PCA):

- We used the first 8 components (correlated to the variables `n_injured_prop`, `n_arrested_prop`, `n_killed_prop`, `n_unharmed_prop`, `n_teen_prop` and `avg_age`)
- The best value of K was 4
- The clustering scores were comparable to that obtained on raw data (the average silhouette score slightly decreased to 0.324)



Hierarchical Clustering



Agglomerative hierarchical clustering scores and characteristics for different linkage methods.

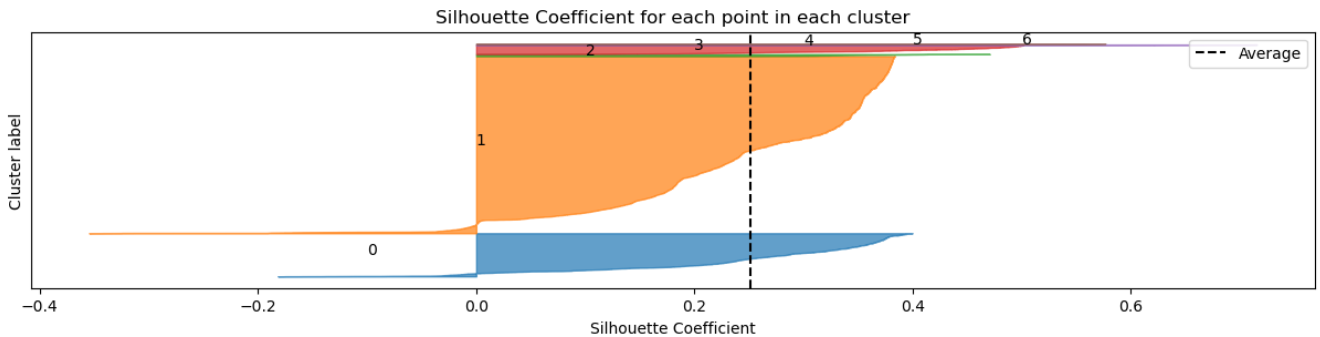
Method	Cut height	Merging difference	# Clusters	Cluster Sizes	CCC	Silhouette Score
single	0.921	0.032	3	0: 13229; 1: 1; 2: 1	0.790	*0.318*
complete	2.108	0.095	10	0: 5637; 2: 2554; 3: 1695; 4: 1190; 1: 833; 5: 715; 6: 470; 8: 59; 9: 57; 7: 21	0.721	0.326
average	1.746	0.106	3	0: 13223; 2: 5; 1: 3	**0.833**	0.331
ward	24.665	3.480	7	2: 4188; 0: 4145; 3: 1806; 5: 1119; 1: 915; 4: 675; 6: 383	*0.699*	**0.368**

- We searched for the optimal cut by calculating the **Silhouette** score for each clustering obtained by cutting the tree at the 10 merging steps with the greatest distance between merged clusters
- **Ward linkage** provides the most balanced outcomes, both in terms of tree structure and cluster sizes

DBSCAN (1)

- Applied to a **subset** of lowly correlated features
- Plot of the distance between each point and its k^{th} nearest neighbor for various values of k (3, 5, 9, 15, 20, 30) to identify the rage for ϵ
- Best configuration chosen based on the silhouette coefficient, the number of clusters, their size, and the percentage of detected noise

eps	min_samples	# Clusters	Noise	% Noise	Silhouette
1.50	**20**	**6**	**168**	**1.269**	**0.251**
2.00	12	3	54	0.408	0.243
2.00	7	4	38	0.287	0.242
1.75	7	4	61	0.461	0.232
1.50	12	5	111	0.839	0.219



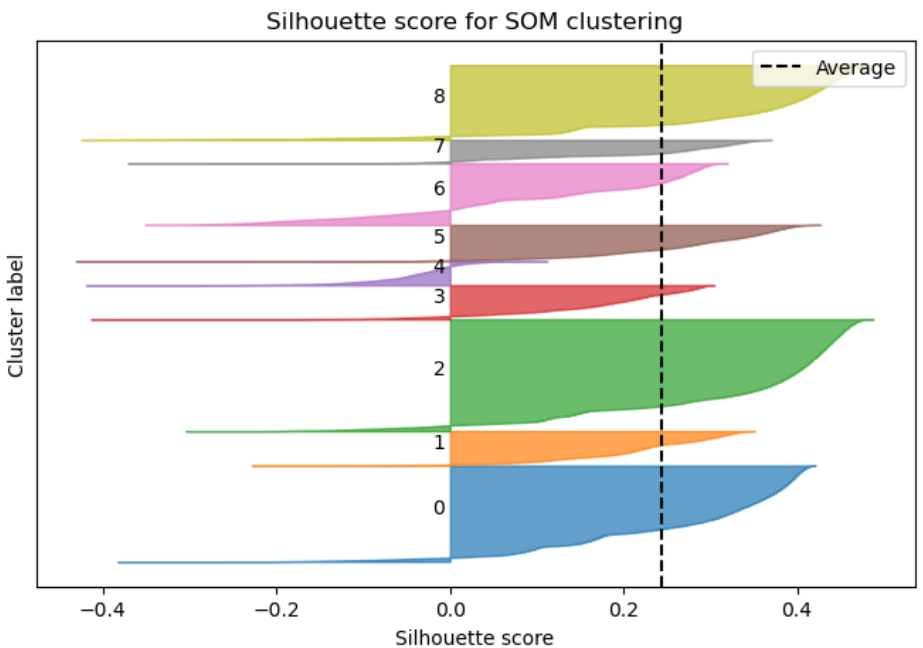
DBSCAN (2)

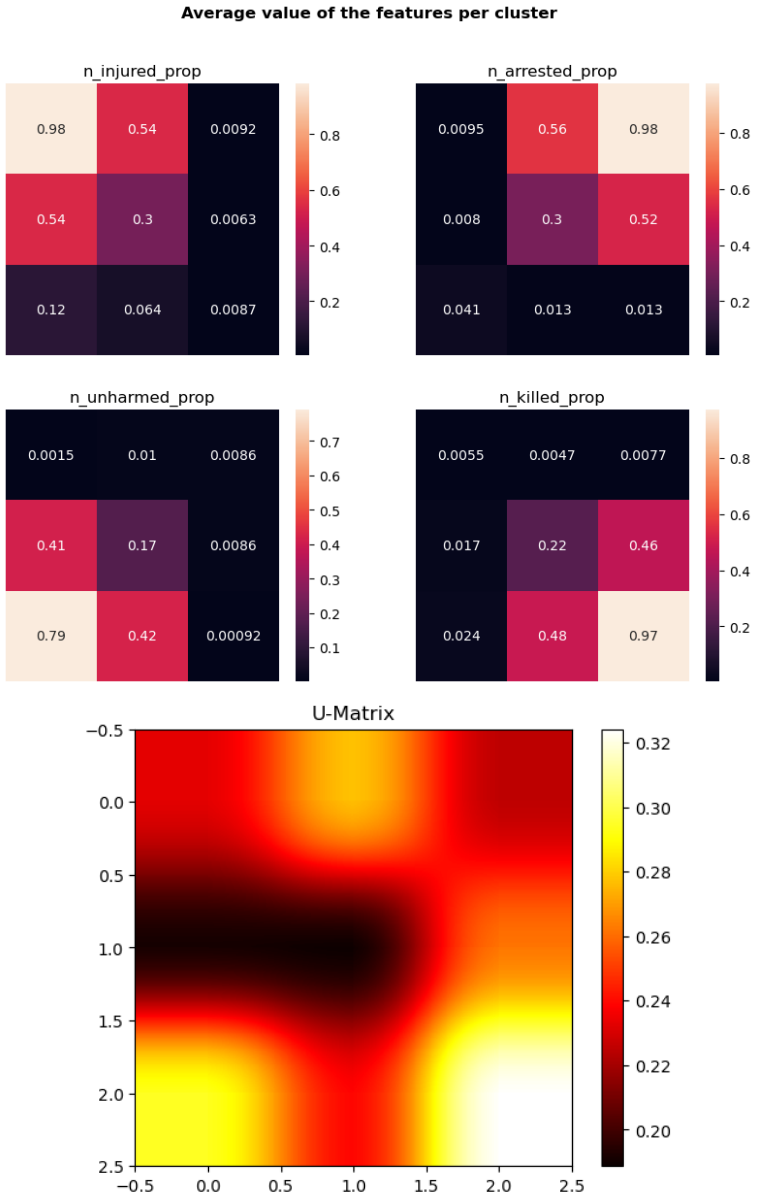
Characterization and interpretation of the obtained clusters:

- Clusters appear to group the data primarily based on the variables `n_killed_prop`, `n_arrested_prop`, and `n_males_prop`
- The variables `n_arrested_prop` and `n_males_prop` exhibit a uniform distribution in clusters 0 and 1, take values close to 0 in clusters 2 and, while in cluster 4 they assume a value of 1
- Among the 168 incidents classified as noise by the algorithm:
  - 113 incidents involve at least one woman
  - for 104 incidents, the average age exceeds 27 (overall average age)
  - 102 incidents involve more than two participants (fourth quartile)
  - all incidents involving a number of participants greater than 6 are included

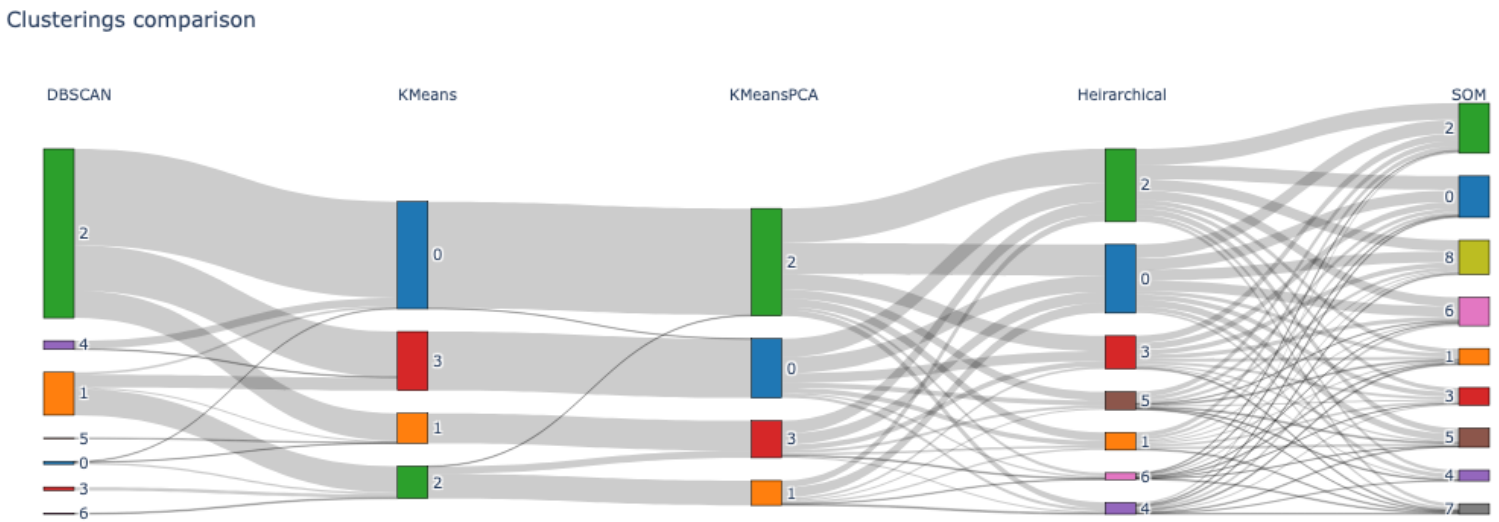
SOM

- We used 3x3 grid (i.e., 9 clusters) with at most 4 neighbors per cell
- Clusters group the data primarily based on the variables `n_injured_prop`, `n_arrested_prop`, `n_unharmed_prop` and `n_killed_prop` (at the corners of the grid)
- The average Silhouette score is 0.244





Clustering Comparison (1)



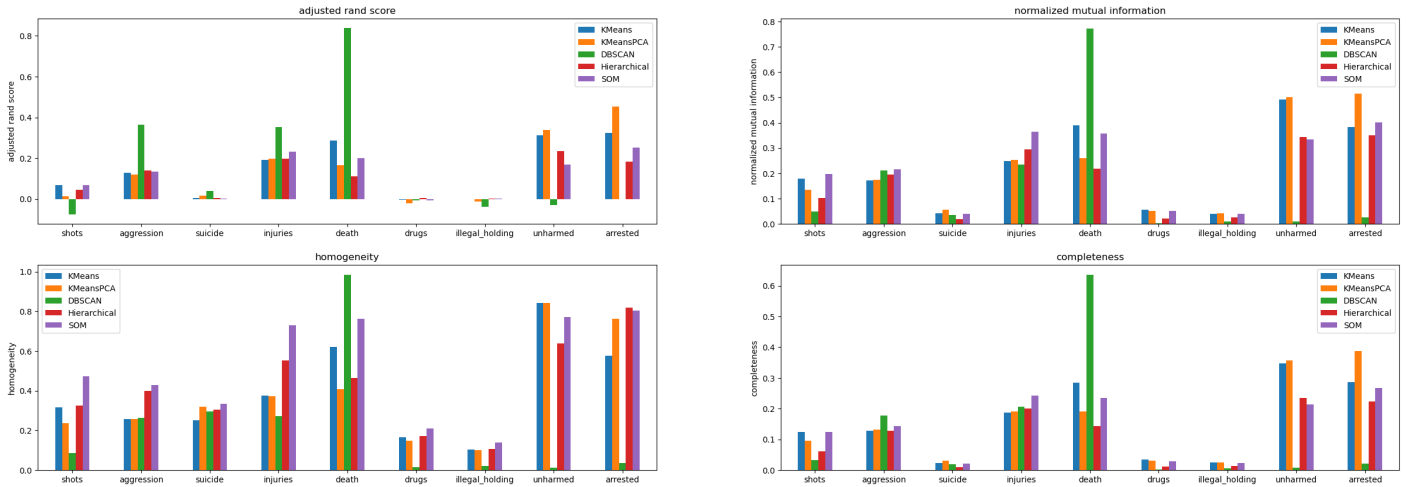
Sankey diagram of the labels assigned by the different clustering algorithms.

Clustering Comparison (2)

Internal index

Algorithm	# Clusters	Silhouette score (std)
K-Means	4	0.327 (-)
K-Means PCA	4	0.324 (-)
Random labeling	4	-0.001 (0126)
DBSCAN	7	0.251 (-)
Random labeling	7	-0.003 (0317)
Ward Hierarchical Clustering	9	**0.368** (-)
Self Organizing Maps	9	*0.244* (-)
Random labeling	9	-0.003 (0420)

External indices



Predictive Analysis

Classification models

We conducted experiments with 10 different classifiers:

- RIPPER
- Decision Trees (DT)
- Random Forest (RF)
- Ada Boost (AB)
- Extreme Gradient Boosting (XGB)
- K-Nearest Neighbors (KNN)
- Nearest Centroid (NC)
- Support Vector Machine (SVM)
- Neural Networks (NN)
- Naive Bayes for mixed data (NB)

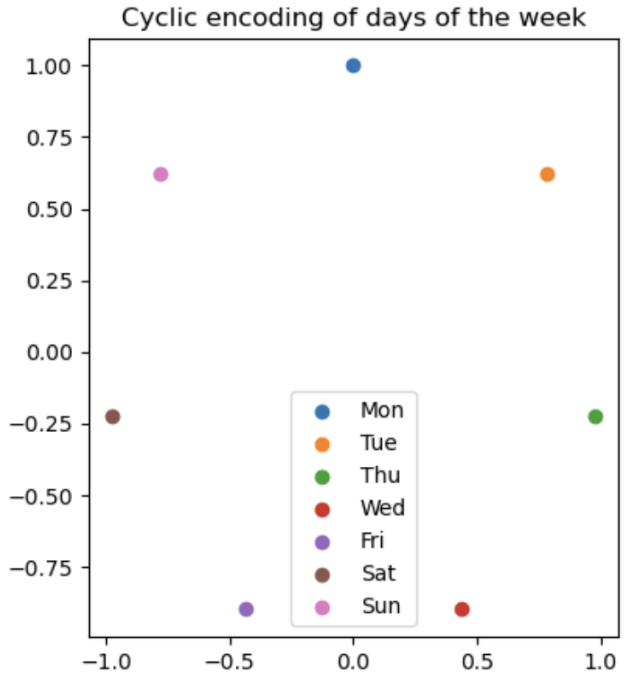
Features - Definition

- We **excluded** features **correlated** to the target variable or that could be partially inferred from the feature `n_participants` and those involving the concept of **surprisal** to prevent any information leakage from the testing set into the training set
- We defined:
  - **Political-economic features**
    `gun_law_rank` (from the Giffords Law Center to Prevent Gun Violence), `poverty_percentage`, `democrat`
  - **Spatial features**
    `x` and `y` (longitude and latitude UTM projections, zone 14)
  - **Temporal features**
    `day`, `day_of_week`, `month`, `year`, `days_from_first_incident`
  - **Incidents characteristics**
    `aggression`, `accidental`, `defensive`, `suicide`, `road`, `house`, `school`, `business`, `illegal_holding`, `drug_alcohol`, `officers`, `organized`, `social_reasons`, `abduction`

Features - Pre-processing

Before applying 'distance-based' classifiers (KNN, NC, SVM and NN), the following steps were performed:

- `day`, `day_of_week` and `month` were **encoded** mapping their values on a circle centered in (0,0) with radius 1
- features were **scaled** in the range [0,1] (fitting Min-Max scaling on each training fold)



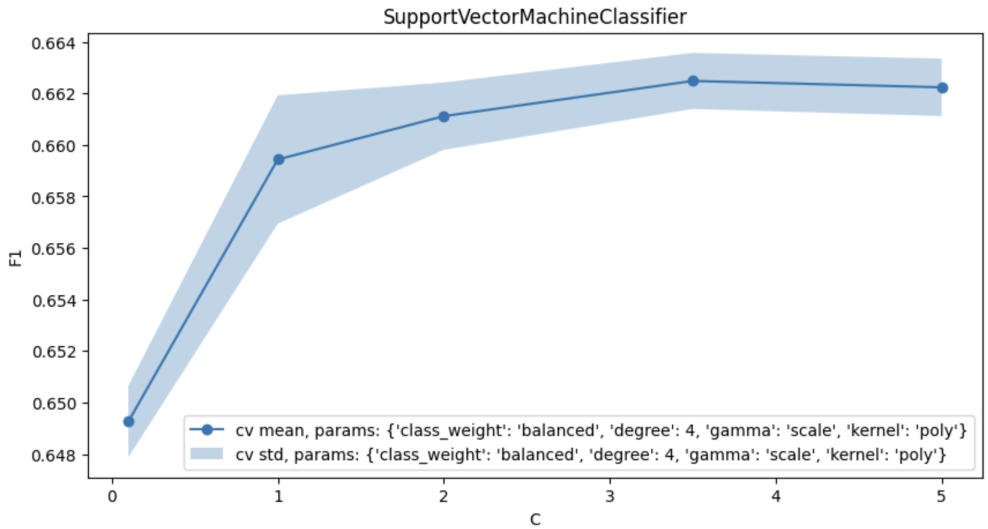
No pre-processing was conducted for the remaining classifiers (RIPPER, DT, RF, AB, XGB, NB).

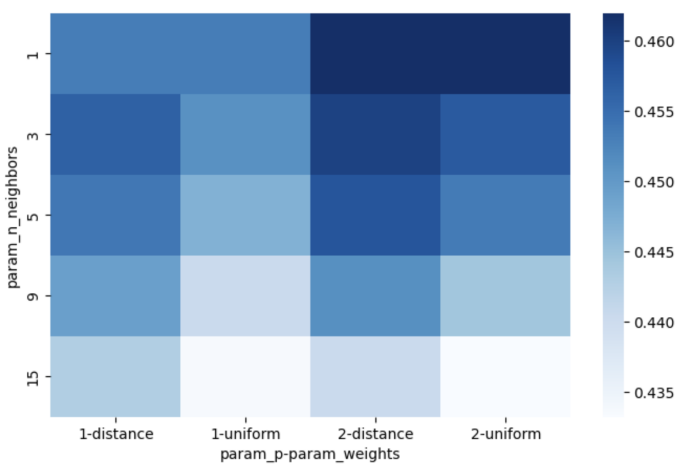
Validation schema

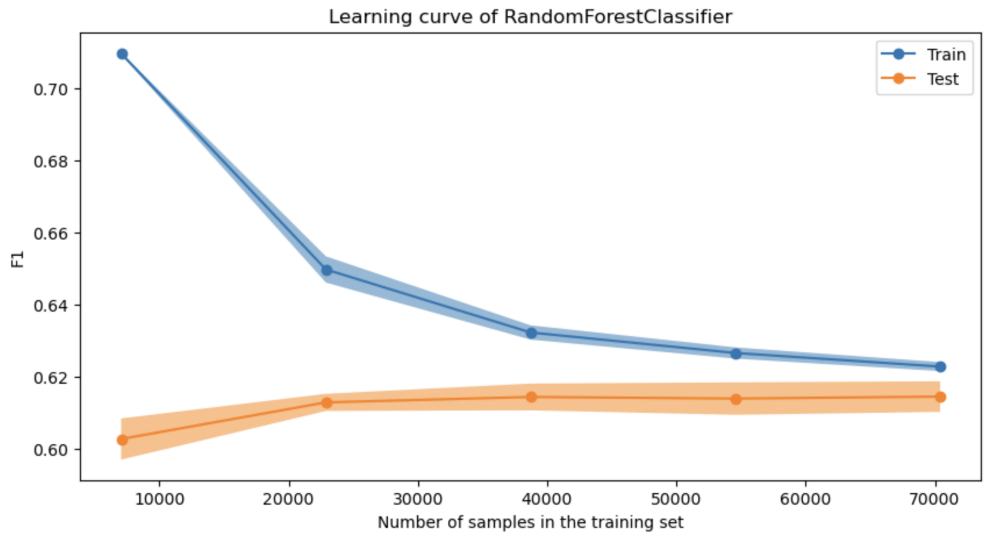
- Data was split stratifying on the target variable into a **development set** (67%) and a **test set** (33%)
- Hyperparameters optimization was conducted through **grid searches**, employing either a stratified 5-fold cross-validation (for models with low training time requirements) or two rounds of stratified splits on the development set, creating training set comprising 67% of the development set
- With Neural Networks, each training set was further split into an internal training set (80%) and an internal validation set (20%) to implement **early stopping**
- For each classifier, the optimal hyperparameter configuration was chosen based on the mean of the **macro-average F1 scores** achieved across the validation folds
- The best models were **re-trained** on the entire development set and **evaluated on the test set**

Performance evaluation

- Heatmaps to study interactions between hyperparameters
- Performance vs. model complexity
- Learning curves
- Decision boundaries



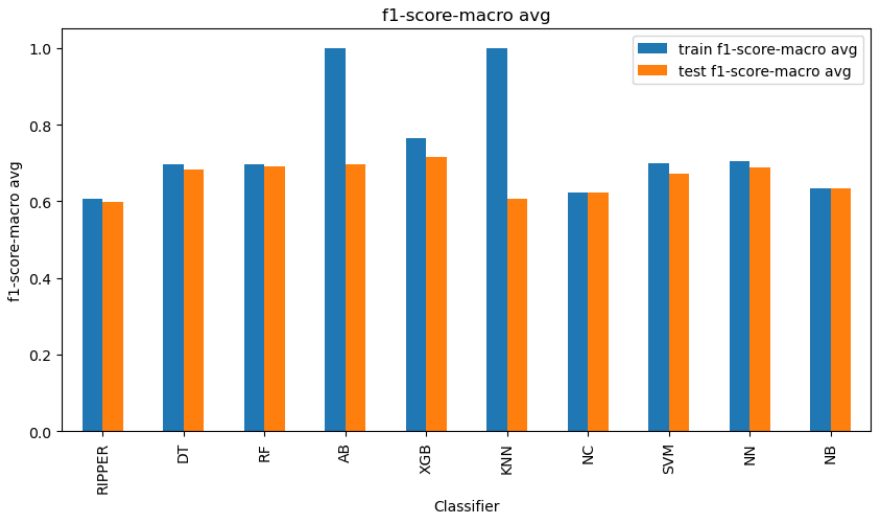




Classification result (1)

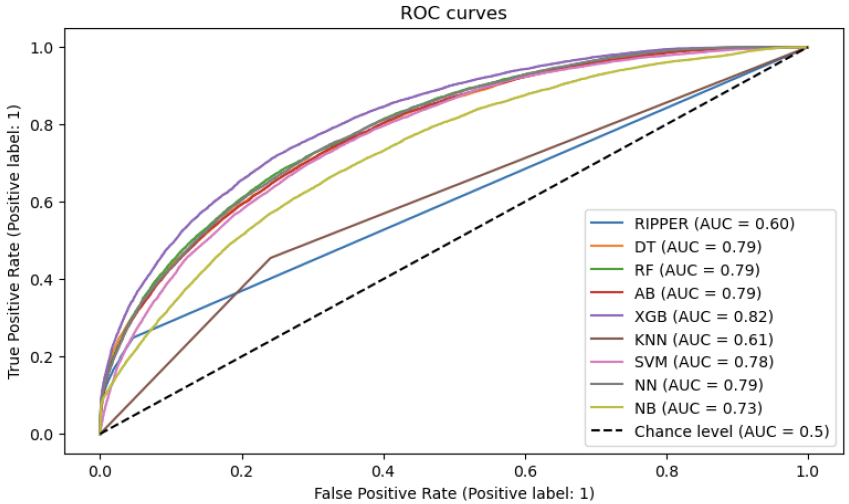
Classifier	Precision Non-Fatal	Recall Non-Fatal	Precision Fatal	Recall Fatal	F1 Score Macro Avg	Accuracy	Auroc
Ripper	0.729	0.954	0.716	0.249	0.598	0.728	0.603
Decision Tree	0.84	0.693	0.524	0.72	0.683	0.701	0.788
Random Forest	0.842	0.707	0.535	0.718	0.691	0.710	0.794
Ada Boost	0.805	0.807	0.588	0.584	0.696	0.736	0.786
Extreme Gradient Boosting	0.853	0.74	0.569	0.729	0.716	0.736	0.818
K Nearest Neighbors	0.747	0.759	0.471	0.455	0.608	0.662	0.607
Nearest Centroid	0.799	0.629	0.457	0.663	0.623	0.640	nan
Support Vector Machine	0.846	0.655	0.505	0.747	0.671	0.685	0.776
Neural Network	0.782	0.873	0.642	0.483	0.689	0.749	0.792
Naive Bayes	0.749	0.879	0.593	0.375	0.634	0.718	0.734

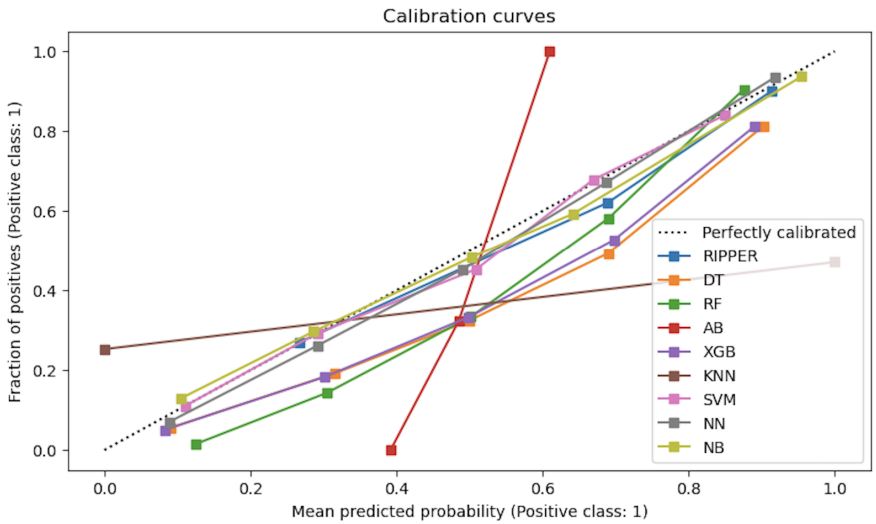
- **XGB** achieves the highest Macro-average F1 score, Ripper achieves the worst performance
- All the classifiers are biased towards the **majority** class
- KNN and AB exhibit a notable *decrease* in performance between training and test set



Classification results (2)

- The highest AUROC score is achieved by XGB
- All the classifiers demonstrate performance superior to random guessing





- The better calibrated models are SVM, NN, NB and RIPPER
- The worst calibrated models are AB and KNN
- AB tends to predict a probability around 0.5 for all instances, while KNN around 0 or 1

Addressing class imbalance

- Using the parameter `class_weight` to train models
- **Random oversampling** (RO)
- **SMOTENC** oversampling

Set	Fatal (%)	Non-Fatal (%)
Training	28175 (32.01)	59835 (67.99)
Training RO	39890 (40.00)	59835 (60.00)
Training SMOTE	39890 (40.00)	59835 (60.00)
Test	14087 (32.01)	29918 (67.90)

Classifier	Precision Non-Fatal	Recall Non-Fatal	Precision Fatal	Recall Fatal	F1 Score Macro Avg
Ripper (Original)	0.729000	0.954000	0.716000	0.249000	0.598000
Ripper (Oversampled)	0.774000	0.847000	0.592000	0.474000	0.667000
Ripper (SMOTE)	0.737000	0.942000	0.698000	0.285000	0.616000
Decision Tree (Original)	0.840000	0.693000	0.524000	0.720000	0.683000
Decision Tree (Oversampled)	0.836000	0.701000	0.527000	0.707000	0.683000
Decision Tree (SMOTE)	0.815000	0.764000	0.557000	0.631000	0.690000
Random Forest (Original)	0.842000	0.707000	0.535000	0.718000	0.691000
Random Forest (Oversampled)	0.842000	0.707000	0.536000	0.718000	0.691000
Random Forest (SMOTE)	0.819000	0.775000	0.571000	0.637000	0.699000
Ada Boost (Original)	0.805000	0.807000	0.588000	0.584000	0.696000
Ada Boost (Oversampled)	0.805000	0.809000	0.591000	0.584000	0.697000
Ada Boost (SMOTE)	0.802000	0.821000	0.599000	0.569000	0.698000
Extreme Gradient Boosting (Original)	0.852000	0.735000	0.564000	0.728000	0.712000
Extreme Gradient Boosting (Oversampled)	0.879000	0.645000	0.51900	0.811000	0.688000
Extreme Gradient Boosting (SMOTE)	0.858000	0.719000	0.556000	0.747000	0.710000
K Nearest Neighbors (Original)	0.747000	0.759000	0.471000	0.455000	0.608000
K Nearest Neighbors (Oversampled)	0.747000	0.759000	0.471000	0.455000	0.608000
K Nearest Neighbors (SMOTE)	0.754000	0.741000	0.470000	0.487000	0.613000
Nearest Centroid (Original)	0.799000	0.629000	0.457000	0.663000	0.623000
Nearest Centroid (Oversampled)	0.797000	0.633000	0.458000	0.658000	0.623000
Nearest Centroid (SMOTE)	0.809000	0.601000	0.452000	0.698000	0.619000
Support Vector Machine (Original)	0.846000	0.655000	0.505000	0.747000	0.671000
Support Vector Machine (Oversampled)	0.847000	0.654000	0.505000	0.749000	0.670000
Support Vector Machine (SMOTE)	0.827000	0.706000	0.524000	0.687000	0.678000
Neural Network (Original)	0.782000	0.873000	0.642000	0.483000	0.689000
Neural Network (Oversampled)	0.782000	0.874000	0.644000	0.483000	0.689000
Neural Network (SMOTE)	0.775000	0.895000	0.668000	0.447000	0.683000
Naive Bayes Mixed (Original)	0.749000	0.879000	0.593000	0.375000	0.634000
Naive Bayes Mixed (Oversampled)	0.778000	0.799000	0.547000	0.515000	0.659000
Naive Bayes Mixed (SMOTE)	0.781000	0.79700	0.549000	0.524000	0.663000

- Performance **does not improve** significantly
- XGB performs better when trained on the original data

Explainability

Explained instances

We explained predictions for two specific incidents in the test set:

- the one with the **highest number of individuals killed**
- one where the variable `suicide` is True but is labeled as 'Non-Fatal' (indicating an **attempted suicide**)

Additionally, for each model, we selected instances predicted as **'Fatal' with the highest probability** and those predicted as **'Non-Fatal' with the highest probability**.

Transparent models

We experimented with two transparent models:

- Explainable Boosting Machine (EBM)
- TabNet (TN)

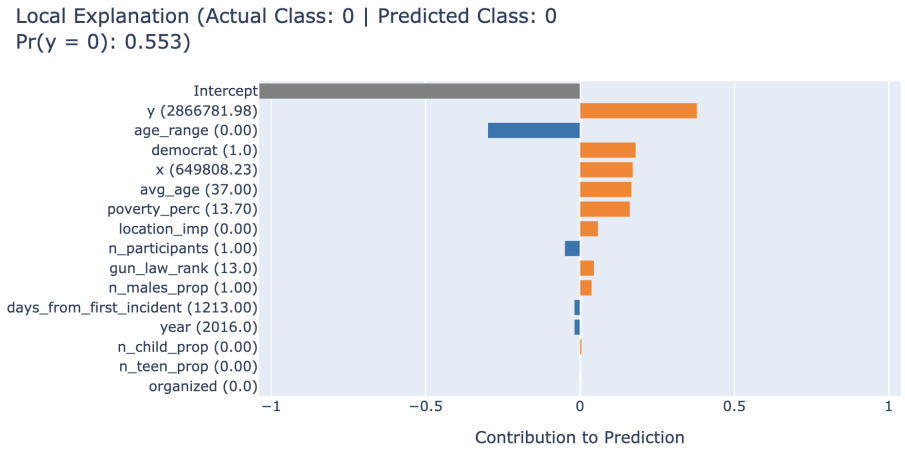
Classifier	Set	Precision Non-Fatal	Recall Non-Fatal	Precision Fatal	Recall Fatal	F1 score Macro Avg	Accuracy	Auroc
EBM	Train	0.779	0.904	0.690	0.455	0.692	0.760	0.804
EBM	Test	0.774	0.903	0.681	0.440	0.684	0.755	0.799
TN	Train	0.764472	0.920	0.702	0.398	0.671	0.753	0.800
TN	Test	0.763325	0.921	0.700	0.394	0.669	0.752	0.795

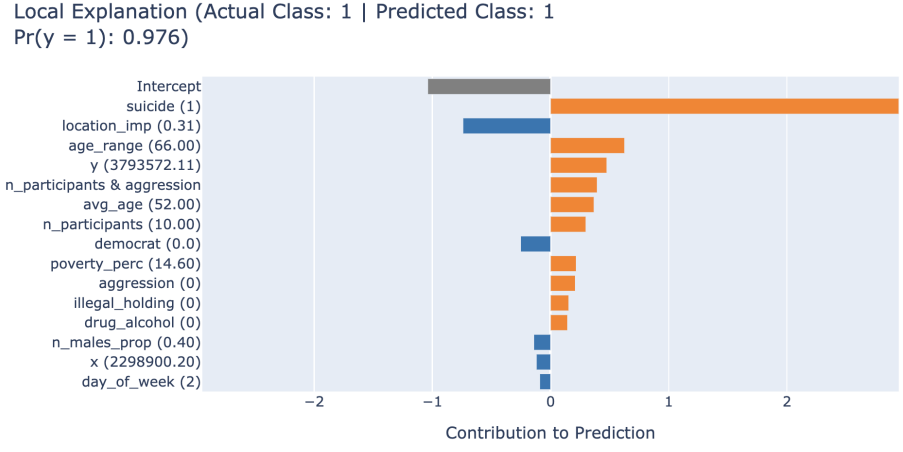
Their performance is **comparable** to that of opaque models.

EBM explanations

Attempted suicide instance:

- correctly predicted as 'Non-Fatal'
- the variable `suicide` does not contribute to the prediction
- the latitude projection has the highest positive contribution
- `age_range` has the highest negative contribution



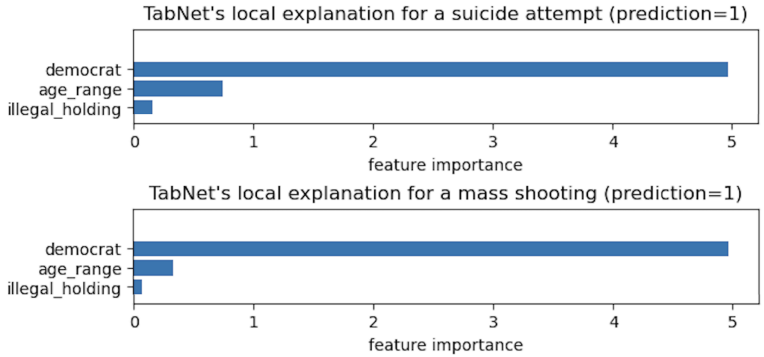


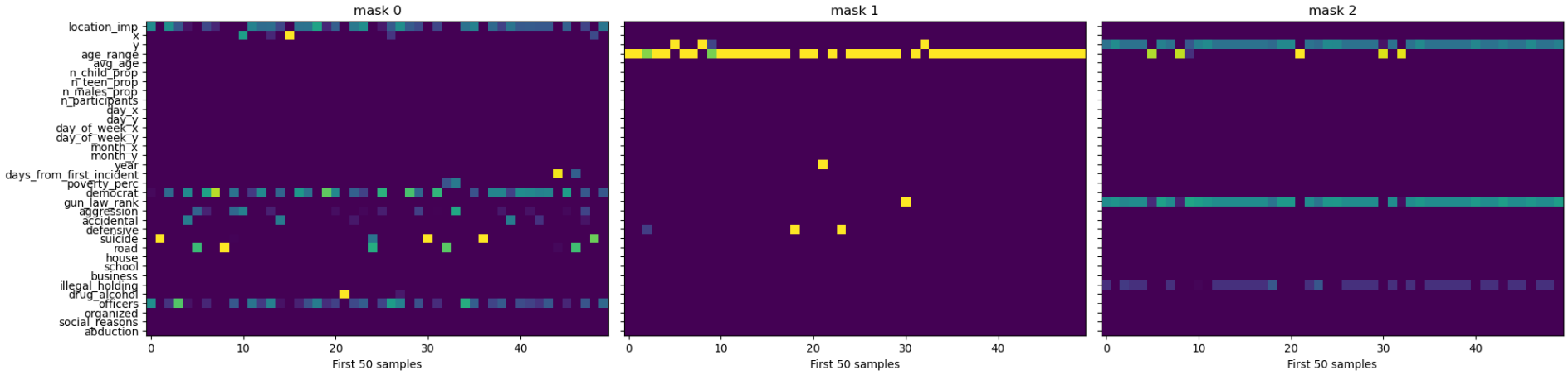
Mass shooting instance:

- correctly predicted as 'Fatal'
- the feature `suicide` exhibits a significantly higher positive contribution compared to other features

TabNet explanations

- TabNet wrongly classifies the attempt suicide incident but correctly classifies the mass shooting incident
- In local explanations the only features that are assigned importances are `democrat`, `age_range` and `illegal_holding`



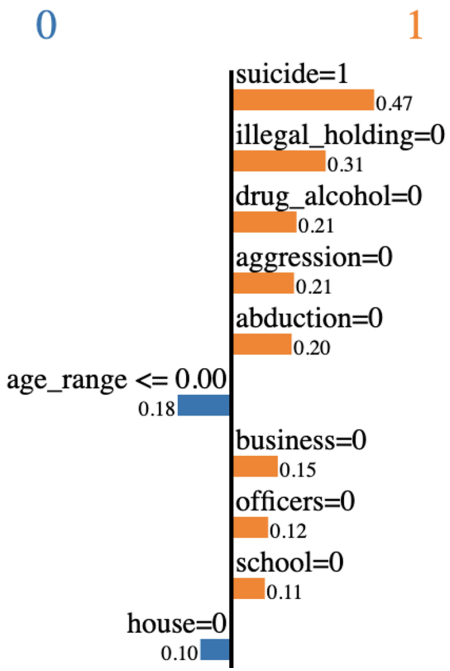


- The first mask focuses `democrat`, `location_imp` and `officers`
- The second mask focuses on `age_range`
- The third mask focuses on `gun_law_rank`, `y` and `illegal_holding`

LIME explanations

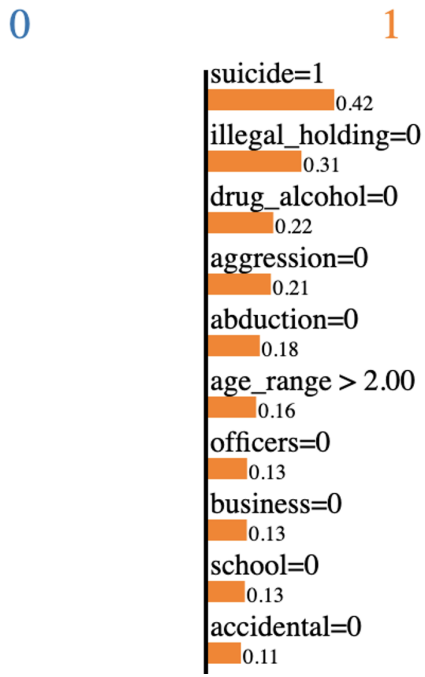
Attempted suicide instance:

- XGB wrongly classifies the incident as 'Fatal' (with a probability of 0.99)
- `suicide` exhibits the highest positive contribution



Mass shooting instance:

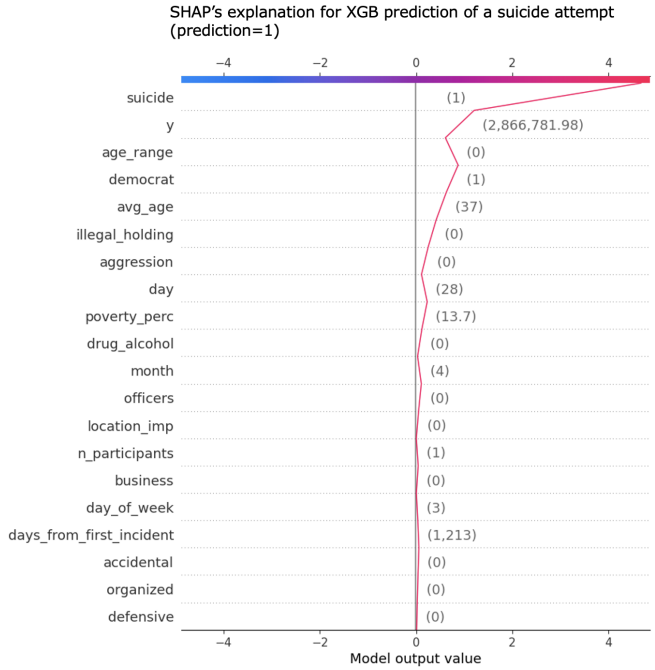
- XGB correctly classifies the incident as 'Fatal' (with a probability of 0.99)
- The five most important features are the same as for the explanation of the prediction for the attempted suicide

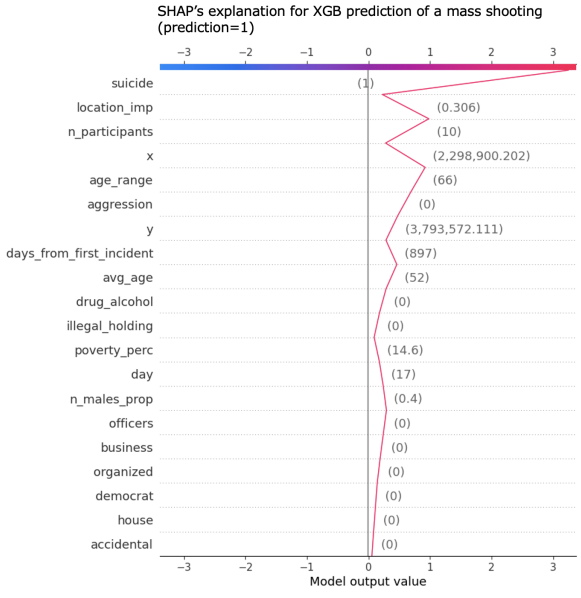


SHAP explanations (1)

Attempted suicide instance:

- XGB wrongly classifies the incident as 'Fatal' (with a probability of 0.99)
- `suicide`, `y` and `democrat` (equal to 1) positively contribute to the prediction



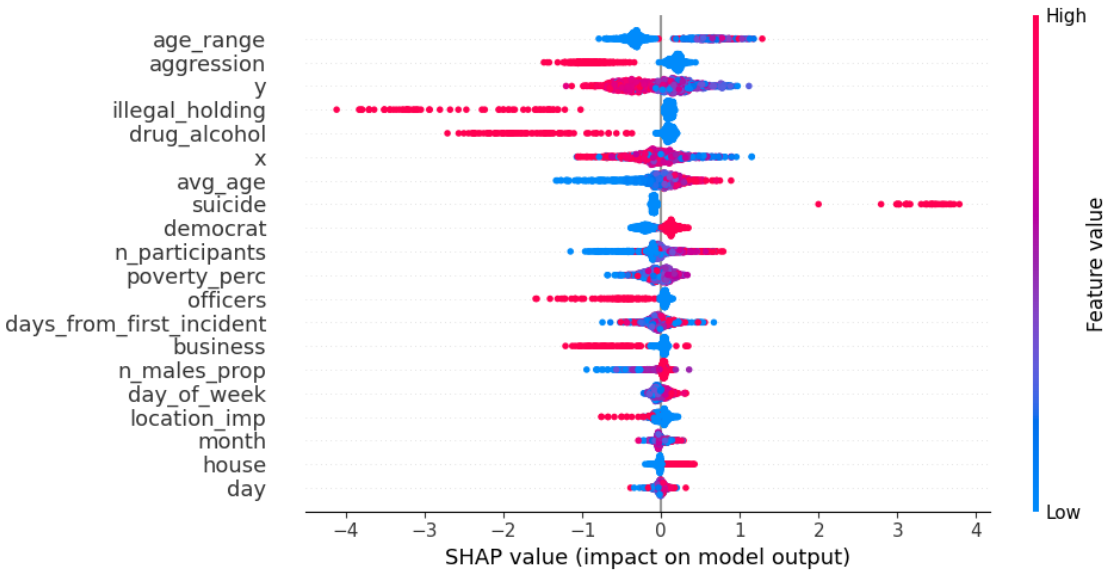


Mass shooting instance:

- XGB correctly classifies the incident as 'Fatal' (with a probability of 0.99)
- The features with the highest positive contribution are `suicide` (equal to 1), `n_participants` (equal to 10), and `age_range` (equal to 66)

SHAP explanations (2)

- `age_range`, `aggression`, `y`, and `illegal_holding` are among the most significant features
- The values of several binary features effectively *distinguish* the samples based on the model's predictions



Explainers comparison

Faithfulness for the LIME and SHAP explanations on different samples.

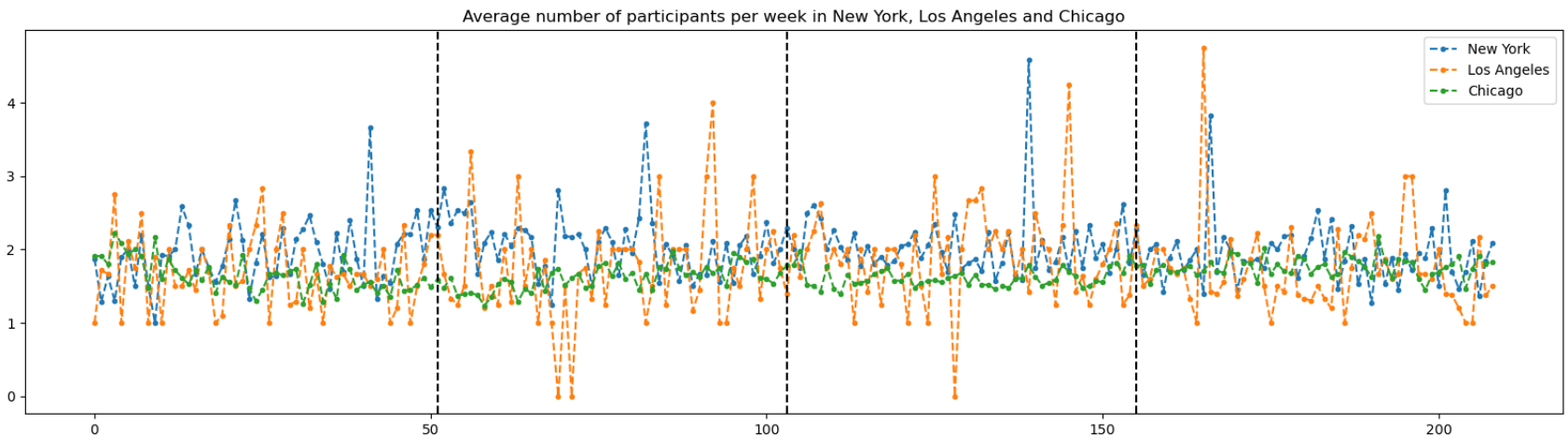
Sample	DT		RF		XGB		NN		SVM	
	LIME	SHAP	LIME	SHAP	LIME	SHAP	LIME	SHAP	LIME	SHAP
Attempted Suicide	0.638216	0.963508	0.604192	0.964190	0.545138	0.979389	-0.049704	0.130451	-0.049129	-0.235384
Mass shooting	0.734065	0.977253	0.682913	0.964476	0.571016	0.918252	-0.212902	-0.037200	-0.037180	0.298565
Fatal with highest confidence for DT	-0.143784	0.762704	0.249907	-0.801124	0.189934	-0.558125	0.050865	-0.442505	0.062525	-0.023103
Fatal with highest confidence for RF	0.715922	0.980199	0.676820	0.971303	0.590276	0.964750	-0.175269	0.126164	-0.133346	0.169740
Fatal with highest confidence for XGB	0.680731	0.966861	0.664125	0.963553	0.599138	0.953075	-0.149152	0.047637	-0.151222	-0.230260
Fatal with highest confidence for NN	0.661191	0.977253	0.695583	0.964476	0.560756	0.918252	-0.219692	-0.033473	-0.019645	0.304121
Fatal with highest confidence for SVM	0.048304	0.349155	0.017421	0.128338	0.032596	0.779718	-0.059077	0.449277	-0.078551	-0.409537
Non-Fatal with highest confidence for DT	0.380304	-0.929084	0.483481	-0.957459	0.412477	-0.965275	0.157702	-0.226454	0.150209	-0.081196
Non-Fatal with highest confidence for RF	0.368766	-0.794060	0.495323	-0.951173	0.388899	-0.955820	0.144798	-0.181697	0.210072	-0.143239
Non-Fatal with highest confidence for XGB	-0.391588	0.697630	0.541675	-0.877779	0.431065	-0.887615	-0.016043	0.074321	0.168056	-0.111436
Non-Fatal with highest confidence for NN	0.304621	-0.711646	0.432901	-0.812596	0.436162	-0.908356	0.072044	-0.018091	-0.089336	0.051093
Non-Fatal with highest confidence for SVM	0.113793	-0.840387	0.139607	-0.773380	0.139134	-0.853630	0.051220	-0.052365	0.097364	-0.186853

- Features 'base values' were computed using the *median* value
- SHAP faithfulness for the explanations of XGB predictions on both the attempted suicide incident and the mass shooting incident is significantly higher than the faithfulness of LIME explanations
- This trend of SHAP *overperforming* LIME on the two instances is consistent across various classifiers
- For instances predicted as 'Fatal' with the highest confidence, SHAP's explanations achieve higher scores, whereas for instances predicted as 'Non-Fatal' with the highest confidence, LIME's explanations perform better

Time Series Analysis

Data preparation

- 588 cities in 51 states are model as time series.
- Each data point represents the mean number of participants per incident per week between 2014 and 2017.
- For weeks with no reported incidents, the values were set to zero.
- Incidents for which the value of `n_incidents` was unknown were removed.
- Time series with less than 15% of non-zero values were not considered.



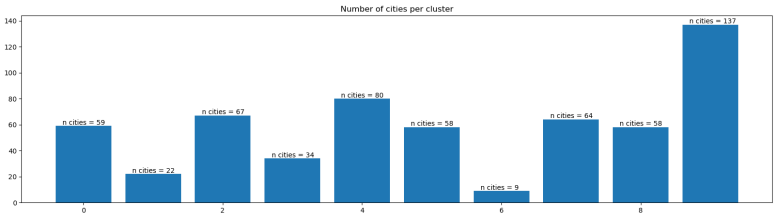
Clustering analysis (1)

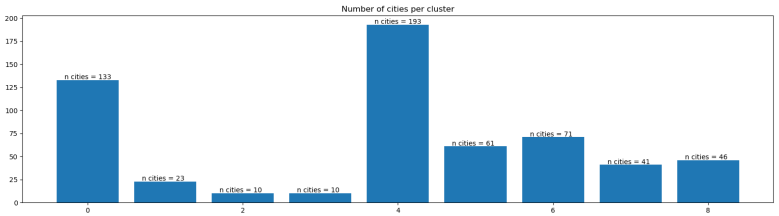
Shape-based clustering:

- `TimeSeriesKMeans` from `tslearn`
- Time series scaled using `TimeSeriesScalerMeanVariance`
- DTW as distance
- K = 10 (elbow method)

Compression-based clustering:

- `TimeSeriesKMeans` from `tslearn`
- PAA representation of time series (from `tslearn`), 100 segments
- DTW as distance
- K = 9 (elbow method)



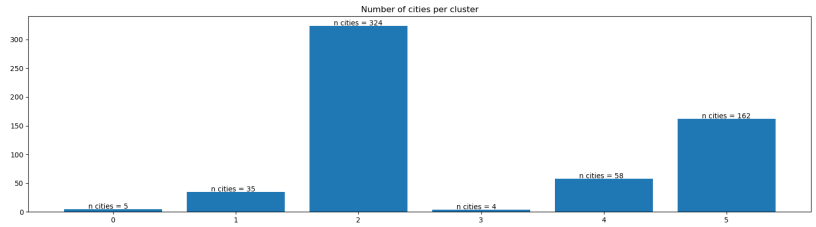


Shaped-based Cluster ID		**Compression-based Cluster ID**
5, 1, 6	Cities with higher frequency of incidents and involving larger number of people.	1, 2, 3, 6
9	Most populated clusters: cities characterized by low number of participants in each incident and a high number of weeks without incidents.	4, 0

Clustering analysis (2)

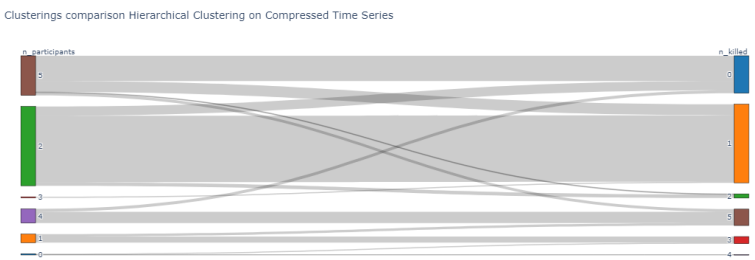
Compression-based clustering:

- `AgglomerativeClustering` from `scikit-learn`
- Pairwise distances (CDM) compression
- Average linkage
- 6 clusters (dendrogram inspection)



	Cluster ID
The most populated cluster: a significant portion of cities with a high frequency of incidents and involving a larger number of people compared to the average.	2
Cities with higher frequency of incidents and involving large number of people.	5

Clusters formed using **time series** generated based on the number of participants are **correlated** with the ones **generated using the number of killed people**.



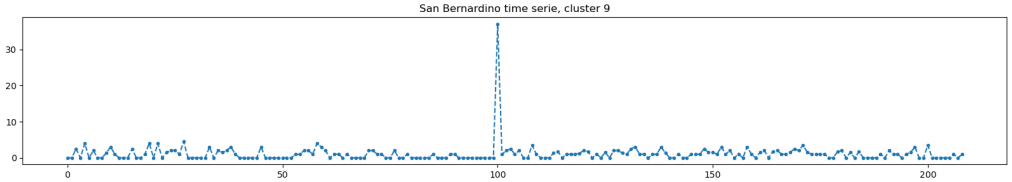
Clustering analysis (3)

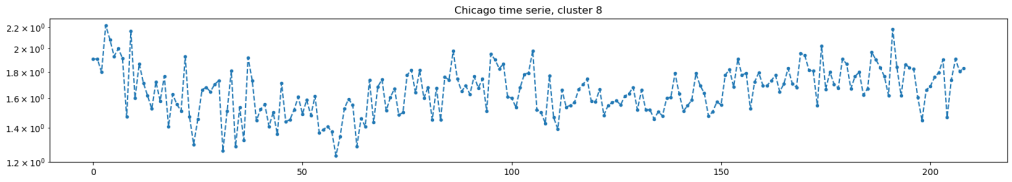
Features-based clustering:

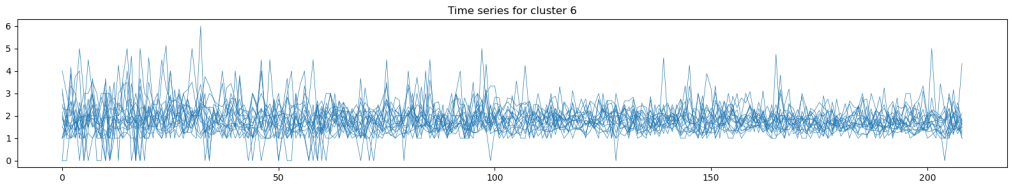
- `KMeans` from `sklearn`
- Features:

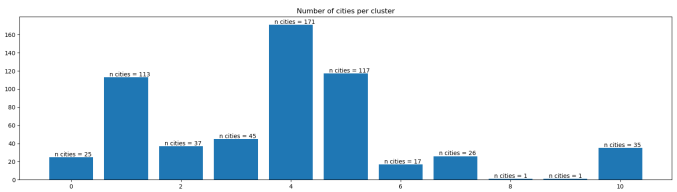
  average of the values
  standard deviation
  percentiles
  median
  IQR
  skewness
  kurtosis

- Euclidian distance
- K = 11 (elbow method)







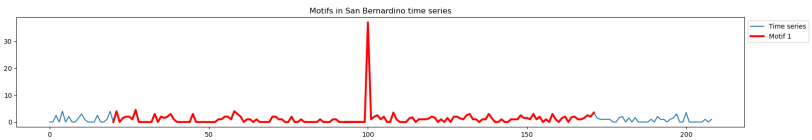


	Cluster ID
High frequency of incidents.	0, 3, **6**, **8**, 10
Low frequency of incidents.	1, 2, **4**, 5, 7

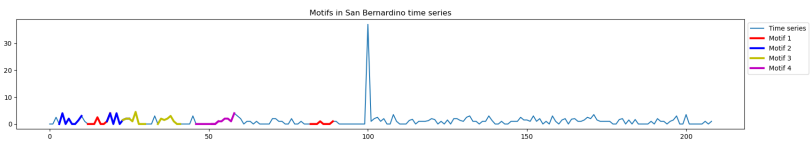
Motifs and anomalies extraction

We used the Matrix Profile algorithm from `matrixprofile-ts` (`motifs` and `discords` methods). The optimal window size was computed using the Highest Autocorrelation method from `claspy`.

San Bernardino time series:

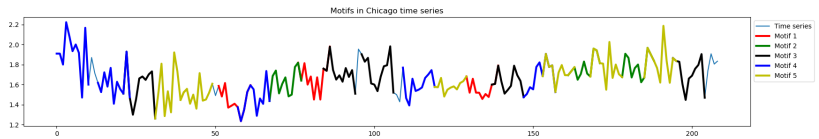


Motifs extraction, optimal window size = 79. Two motifs detected. San Bernardino attack corresponds to an incident took place on July 7, 2014, with 7 participants.

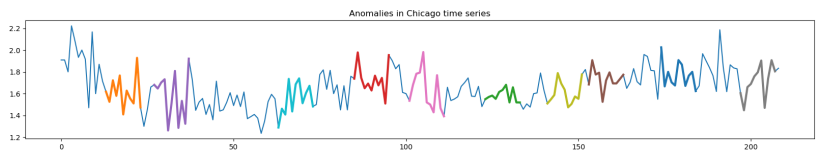


Motifs extraction, window size = 8. Recurrent patterns in the earlier part of the time series.

Chicago time series:



The motif extraction with the optimal window size of 11 reveals 5 different motifs distributed across all time series.



Anomalies detection, optimal window size = 11. The anomalies are detected throughout the entire time series and do not seem to be tied to specific incidents or weeks with a notable frequency of events.

Shapelets extraction and classification (1)

Binary labels for classification:

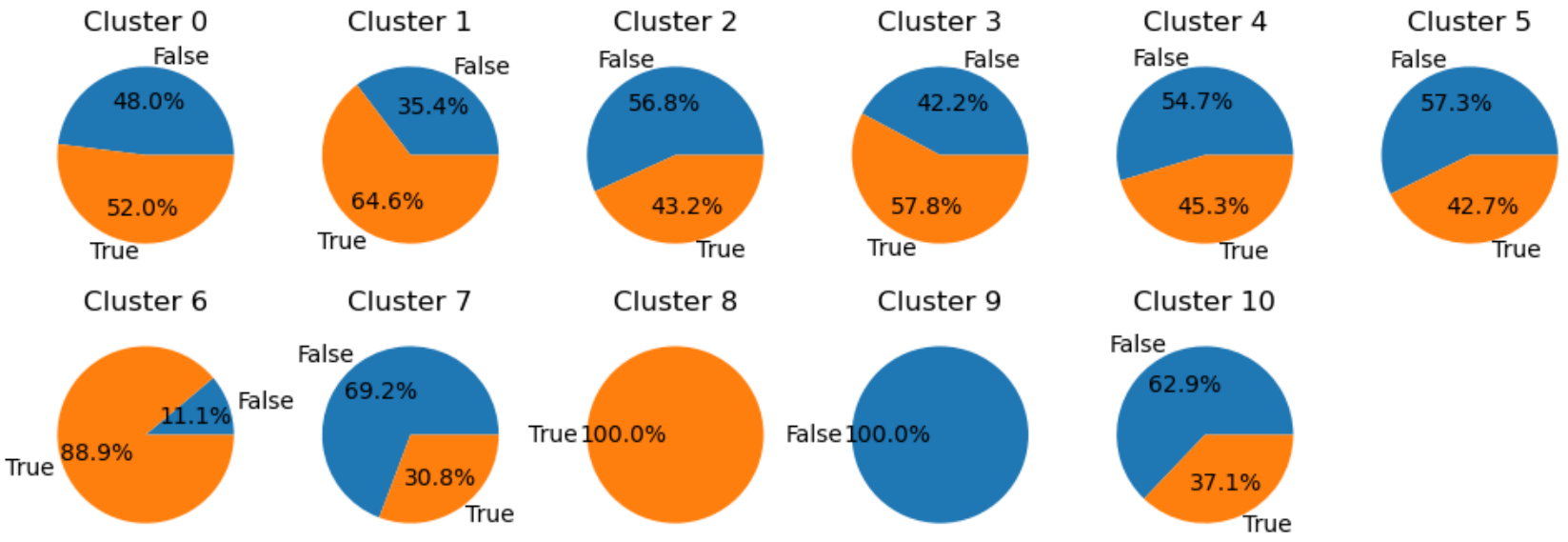
$$\texttt{is\_killed} = \begin{cases} \text{True} & \text{if } \texttt{fatal\_incidents\_ratio} > \\ & \textit{median}(\texttt{fatal\_incidents\_ratio}) \\ \text{False} & \text{otherwise} \end{cases}$$

with `fatal_incidents_ratio` := ratio of fatal incidents in the city

This variable is independent of the number of null values present in the time series, posing a challenging classification task.

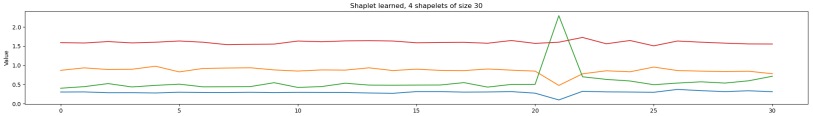
Shapelets extraction and classification (2)

- Dataset divided into training and test sets, with the test set comprising 20% of the total data.
- **Stratification** was applied based on the **clustering memberships** obtained using the **feature-based K-Means** clustering algorithm.
- This ensures a representative distribution of clusters in both the training (49.79% of True labels) and test sets (50% of True labels).



Shapelets extraction and classification (3)

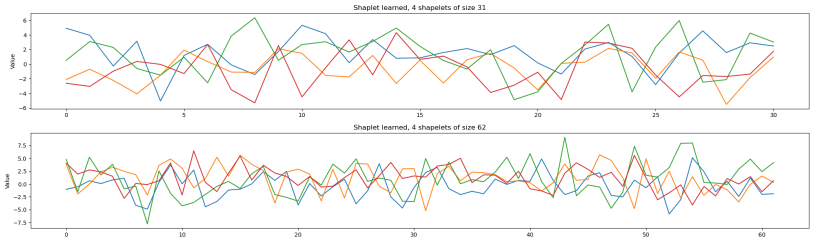
`ShapletModelClassifier`



Both extracts shaplets from time series and also performs the classification task.

Shaplets exhibit relatively **linear patterns**, except for one that displays a **significant peak**, resembling the pattern observed in the time series of San Bernardino.

`DecisionTreeClassifier` and `KNeighborsClassifier`



Extracted shaplets from the time series data using the `LearningShapelets` model.

Shaplets exhibit an **irregular pattern**, but all within the same range of values.

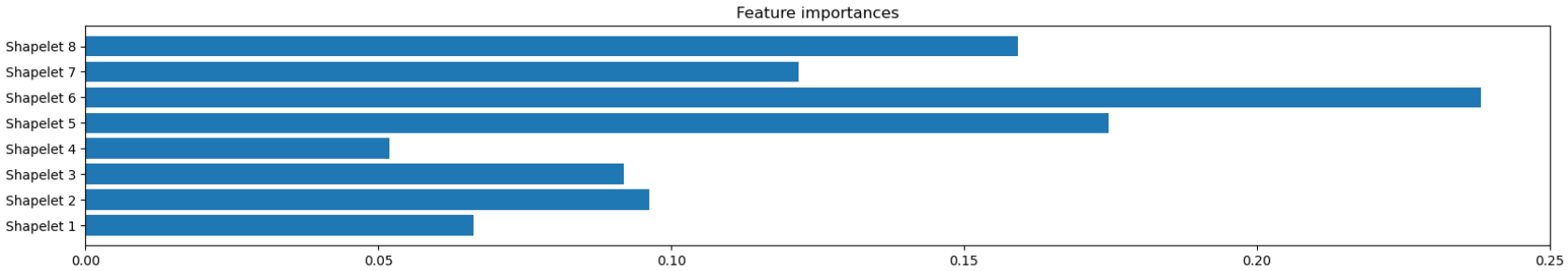
Shapelets extraction and classification (4)

Performance of shaplets classifiers and `RocketClassifier` on test set.

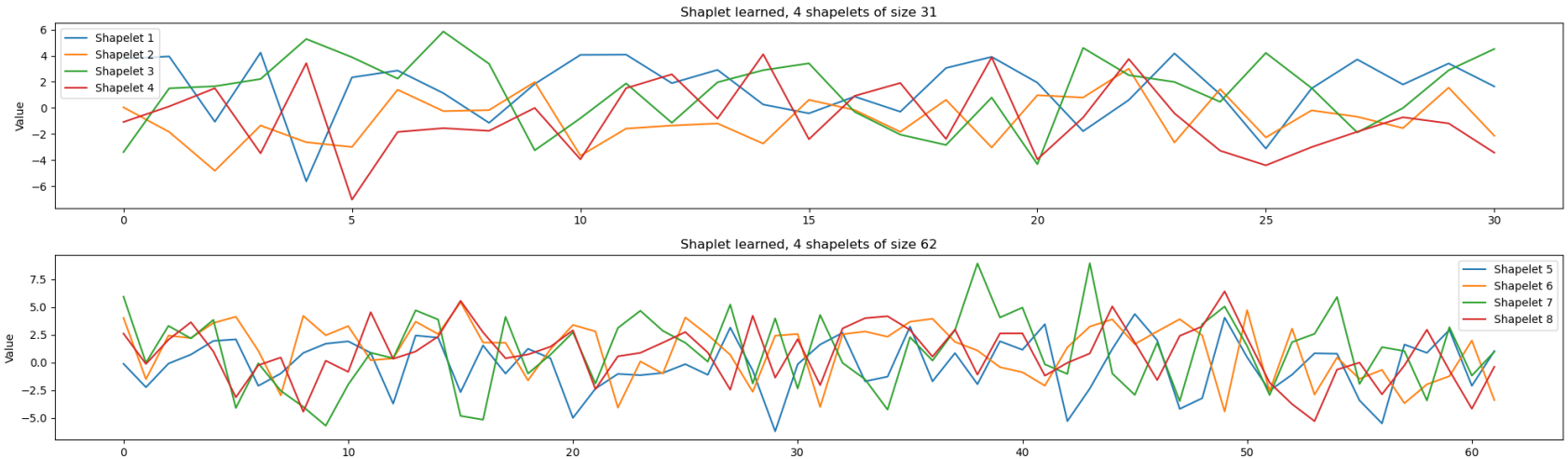
Classifier	Precision False	Recall False	Precision True	Recall True	F1 score macro avg	Accuracy
ShapletModel	0.786	0.373	0.589	0.898	**0.609**	**0.636**
DecisionTreeClassifier	0.607	0.627	0.614	0.593	0.610	0.610
KNeighborsClassifier	0.582	0.542	0.571	0.610	0.576	0.576
RocketClassifier	0.564	0.525	0.556	0.593	0.559	0.559

- All models demonstrated a relatively balanced performance, with trade-offs between precision and recall for both classes.
- DecisionTreeClassifier assigns higher feature importance to the shaplets of larger size.
- RocketClassifier on the time series achieves slightly lower performance compared to all the shaplet classifiers.

Shapelets extraction and classification (5)



Features importance assigned from `DecisionTreeClassifier` to shapelets.



Shaplets extracted using the `LearningShapelets` model, 4 shaplets of size 31, and 4 of size 62.

Supplementary Material

Edit distance

Damerau-Levenshtein distance between two strings s and t

$$D(i,j) = \min \begin{cases} D(i-1,j)+1 \\ D(i,j-1)+1 \\ D(i-1,j-1)+\delta \\ D(i-2,j-2)+\delta \quad \text{if } s[i]=t[j] \text{ and } s[i-1]=t[j-1] \end{cases}$$

where:

- $D(i,j)$ is the Damerau-Levenshtein distance between the first i letters of a string s and the first j letters of a string t
- δ is 0 if the current letters $s[i]$ and $t[j]$ are equal, otherwise, it is 1
- $D(i-2,j-2)+\delta$ represents transposition (swapping two adjacent letters) if the current letters $s[i]$ and $t[j]$ are equal, and the preceding letters $s[i-1]$ and $t[j-1]$ are also equal

Surprisal

The surprisal of an event E with probability p(E) is defined as log(1/p(E)) (or equivalently -log(p(E))). Surprisal is inversly related to probability (hence the term 1/p(E)): when p(E) is close to 1, the surprisal of the event is low, whereas when p(E) is close to 0, the surprisal of the event is high. The log gives 0 surprise when the probability of the event is 1. The surprisal is closely related to entropy, which is the expected value of the information content of a random variable, thus quantifying how surprising the random variable is "on average".

Specifically, surprisals were computed w.r.t. the incidents happened in the same semester of the same year and in the same congressional district of the same state.

Clustering evaluation - Internal indices

Davies-Bouldin score measures the average similarity of each cluster with its most similar cluster, where similarity is the ratio of within-cluster distances to between- cluster distances. The lower the better.

The **Calinski-Harabasz** score measures the ratio of the sum of between-cluster dispersion and of within-cluster dispersion. The higher the better.

The **Cophenetic Correlation Coefficient** measures the correlation between the entries of the cophenetic distance matrix and the dissimilarity matrix. The cophenetic distance between two objects is the proximity at which an agglomerative hierarchical clustering technique puts the objects in the same cluster for the first time.

Clustering evaluation - External indeces

Adjusted Rand Score computes a similarity measure between two clusterings by considering all pairs of samples and counting pairs that are assigned in the same or different clusters in the predicted and true clusterings. It is 0 for random labeling, 1.0 when the clusterings are identical and is bounded below by -0.5 for especially discordant clusterings.

Normalized Mutual Information is a normalized version of the Mutual Information (MI) score to scale the results between 0 (no mutual information) and 1 (perfect correlation). Mutual Information is a function that measures the agreement of the two assignments, ignoring permutations.

Homogeneity measures the degree to which each cluster contains only members of a single class; it ranges between 0 and 1, with 1 denoting perfectly homogeneous labeling.

Completeness score measures the degree to which data points that are members of a given class are also elements of the same cluster; it ranges between 0 and 1, with 1 denoting perfectly complete labeling.

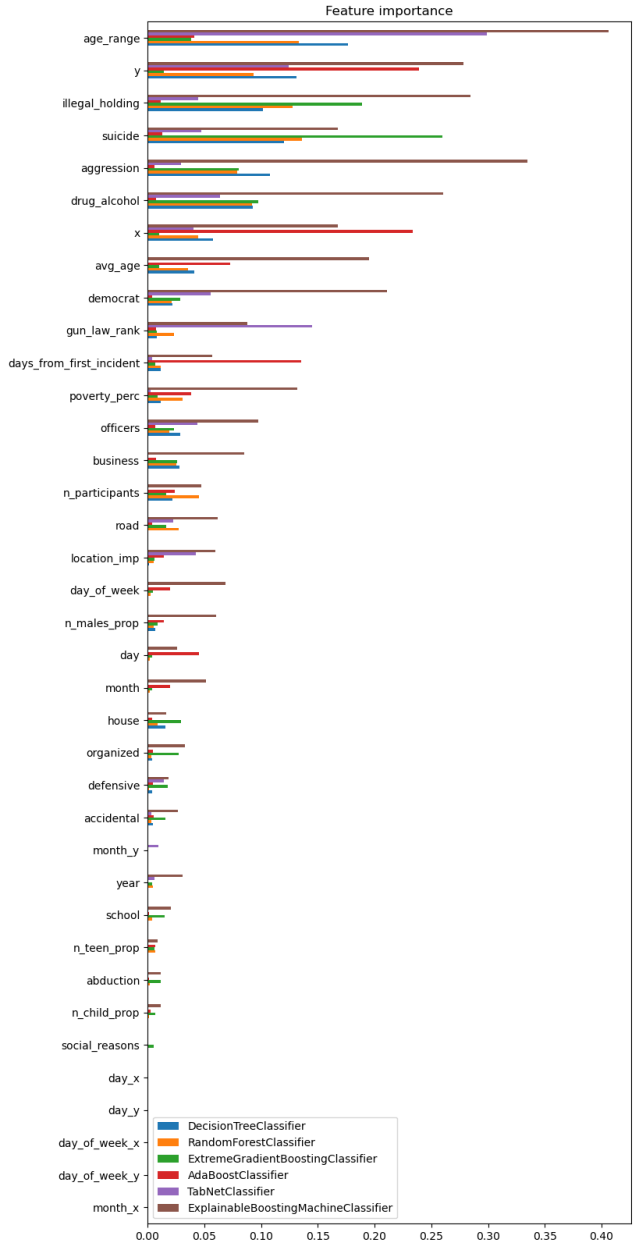
Features for classification - Overview

Name	Description	Rule-based	Distance-based
gun_law_rank	Gun law strength ranking in the state and year where the incident occurred	✓	✓
poverty_perc	Poverty percentage in the state and year where the incident occurred	✓	✓
democrat	Whether the Democratic Party won the last elections in the congressional district of the state and in the year where the incident occurred	✓	✓
location_imp	Location importance according to Geopy	✓	✓
x	Longitude projection using the Universal Transverse Mercator system (UTM)	✓	✓
y	Latitude projection using the Universal Transverse Mercator system (UTM)	✓	✓
day	Day of the month in which the incident occurred	✓	✗
day_x	Projection on the x axis of the day of the month in which the incident occurred	✗	✓
day_y	Projection on the y axis of the day of the month in which the incident occurred	✗	✓
day_of_week	Day of the week in which the incident occurred	✓	✗
day_of_week_x	Projection on the x axis of the day of the week in which the incident occurred	✗	✓
day_of_week_y	Projection on the y axis of the day of the week in which the incident occurred	✗	✓
month	Month in which in the incident occurred	✓	✗
month_x	Projection on the x axis in which the incident occurred	✗	✓
month_y	Projection on the y axis in which the incident occurred	✗	✓
year	Year in which the incident occurred	✓	✓
days_from_first_incident	Days passed since the occurred of the first incident	✓	✓
aggression	Tells if the incident involved an aggression	✓	✓
accidental	Tells if the incident was accidental	✓	✓
defensive	Tells if the incident was defensive	✓	✓
suicide	Tells if the incident was a suicide	✓	✓
road	Tells if the incident occurred in a road	✓	✓
house	Tells if the incident occurred in a house	✓	✓
school	Tells if the incident occurred in a school	✓	✓
business	Tells if the incident occurred at a business	✓	✓
illegal_holding	Tells if the incident involved illegal holding	✓	✓
drug_alcohol	Tells if the incident involved drugs or alcohol	✓	✓
officers	Tells if the incident involved officers	✓	✓
organized	Tells if the incident was carried by an organization or a group	✓	✓
social_reasons	Tells if the incident involved social discriminations or terrorism	✓	✓
abduction	Tells if the incident involved any form of abduction	✓	✓
age_range	Range of age of the participants involved in the incident	✓	✓
avg_age	Average age of the participants involved in the incident	✓	✓
n_child_prop	Proportion of children involved in the incident	✓	✓
n_teen_prop	Proportion of teen involved in the incident	✓	✓
n_males_prop	Proportion of males involved in the incident	✓	✓
n_participants	Number of participants involved in the incident	✓	✓
	Total number of features	31	34

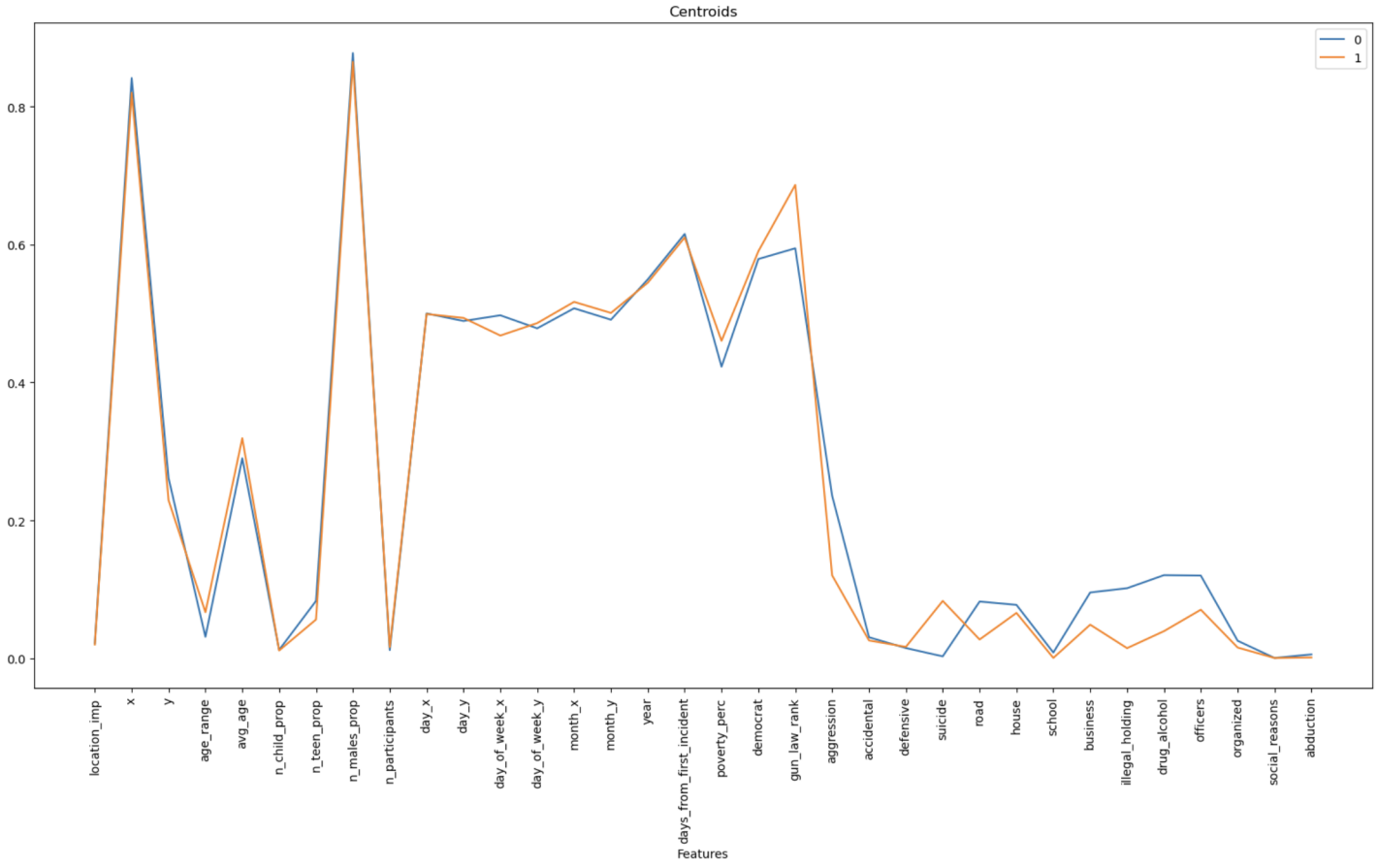
Rule-based Classifiers	Ripper, Decision Trees, Random Forest, Ada Boost, Extreme Gradient Boosting, Naive Bayes
Distance-based Classifiers	K Nearest Neighbors, Nearest Centroid, Support Vector Machine, Neural Networks

Global feature importance

- Feature importance values where obtained from classifiers trained on the development set with optimal hyperparameters (note that not all the models where trained using the same set of features and EBM feature importances do not sum to 1)
- There is limited agreement among feature importance values



Class centroids from NC



The features primarily differing in the centroids are 'aggression', 'gun_law_rank', 'drug_alcohol', 'illegal_holding', 'suicide', 'business', 'road' and 'officers'.

LIME parameters

- The class 'LimeTabularExplainer' was employed with default parameters:
  - exponential kernel with a width equal to $\sqrt{\text{n\_features}} \times 0.75$
  - 'auto' mode for feature selection
  - feature quantile discretization for continuous features
  - points sampled from a normal distribution centered on the mean of each continuous feature
- For each instance to explain:
  - 5000 samples were generated
  - the euclidean distance was used as distance metric
  - a Ridge Regressor was fitted

Explainers monotonicity

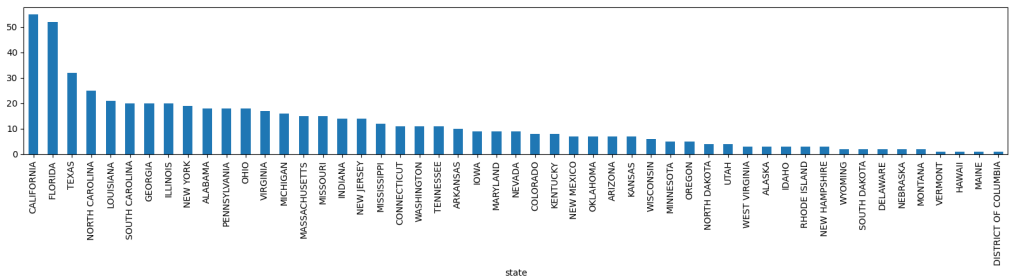
Monotonicity for the LIME and SHAP explanations on different samples.

Sample	DT		RF		XGB		NN		SVM	
	LIME	SHAP	LIME	SHAP	LIME	SHAP	LIME	SHAP	LIME	SHAP
Attempted Suicide	True	True	False	False	False	False	True	True	False	False
Mass shooting	True	True	False	False	False	False	True	False	False	False
Fatal with highest confidence by DT	False	False	False	False	False	False	False	False	False	False
Fatal with highest confidence by RF	True	True	False	False	False	False	True	True	False	False
Fatal with highest confidence by XGB	False	True	False	False	False	False	True	False	False	False
Fatal with highest confidence by NN	True	True	False	False	False	False	True	False	False	False
Fatal with highest confidence by SVM	False	False	False	False	False	False	True	True	False	False
Non-Fatal with highest confidence by DT	False	True	False	False	False	False	False	False	False	False
Non-Fatal with highest confidence by RF	False	True	False	False	False	False	False	False	False	False
Non-Fatal with highest confidence by XGB	False	True	False	False	False	False	False	False	False	False
Non-Fatal with highest confidence by NN	False	False	False	False	False	False	False	False	False	False
Non-Fatal with highest confidence by SVM	True	True	False	False	False	False	False	False	False	False

- Monotonicity is guaranteed only with the explanations for the predictions made DT and NN
- DT exhibits greater monotonicity in SHAP explanations compared to LIME, whereas NN demonstrates higher monotonicity in LIME explanations than SHAP

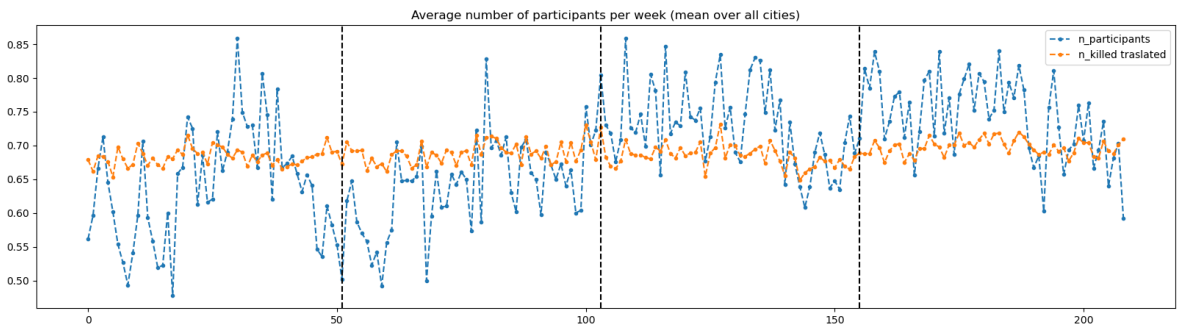
Time series

Time series were generated to represent the mean number of participants per incident per week in each city. The time series assumes value 0 when there are no incidents in the week.



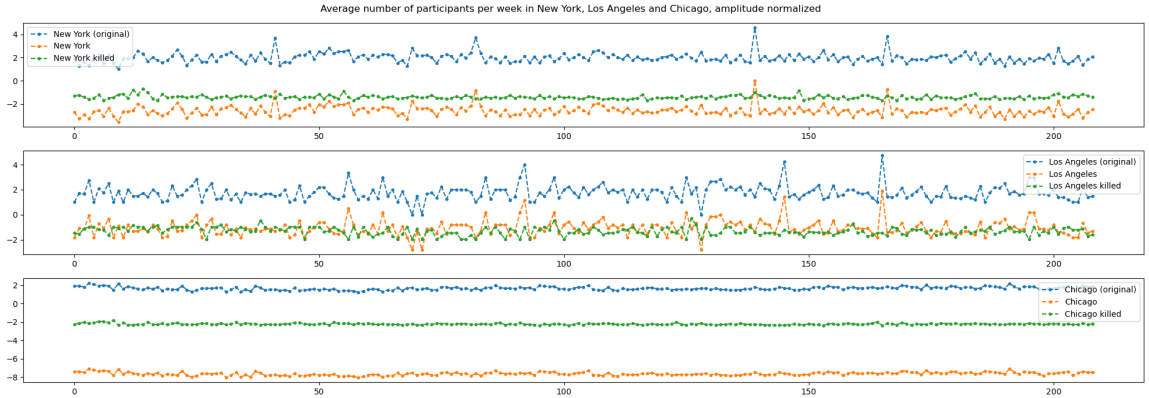
We have at least one time series for each of the 51 states. The states with the highest number of cities modeled as time series are California and Florida.

Time series were generated in the same manner as the previous ones, but this time, using the number of killed people. This was done to compare clustering results between the two datasets.



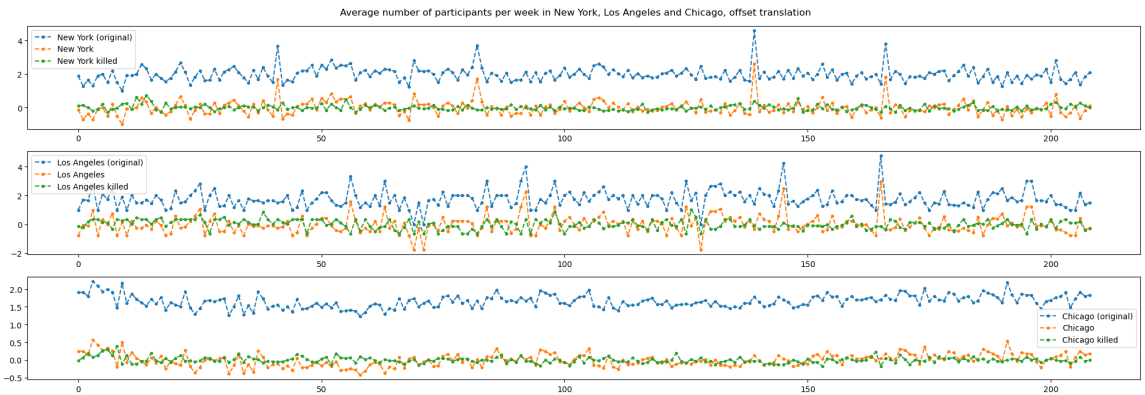
In the image, the time series corresponding to the average of the time series formed with `n_killed` is shifted to have the same mean as the time series of `n_participants`.

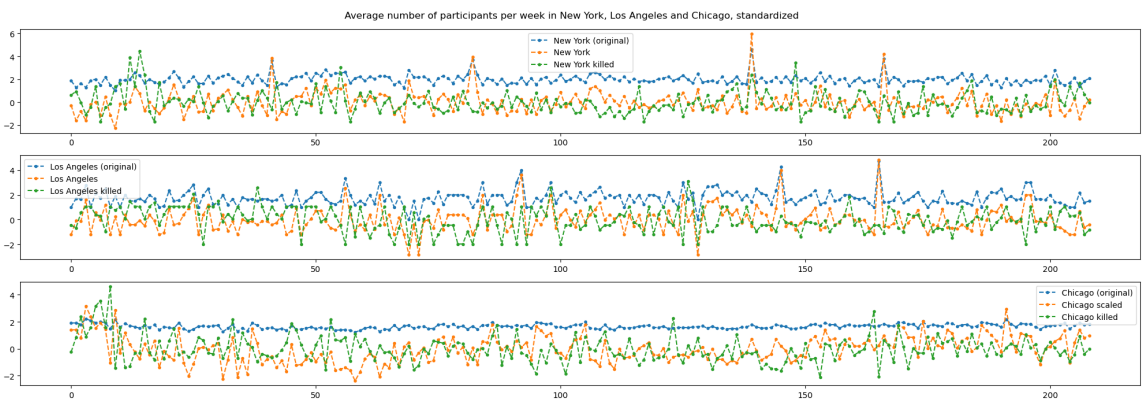
Time series scaling



Amplitude Scaling

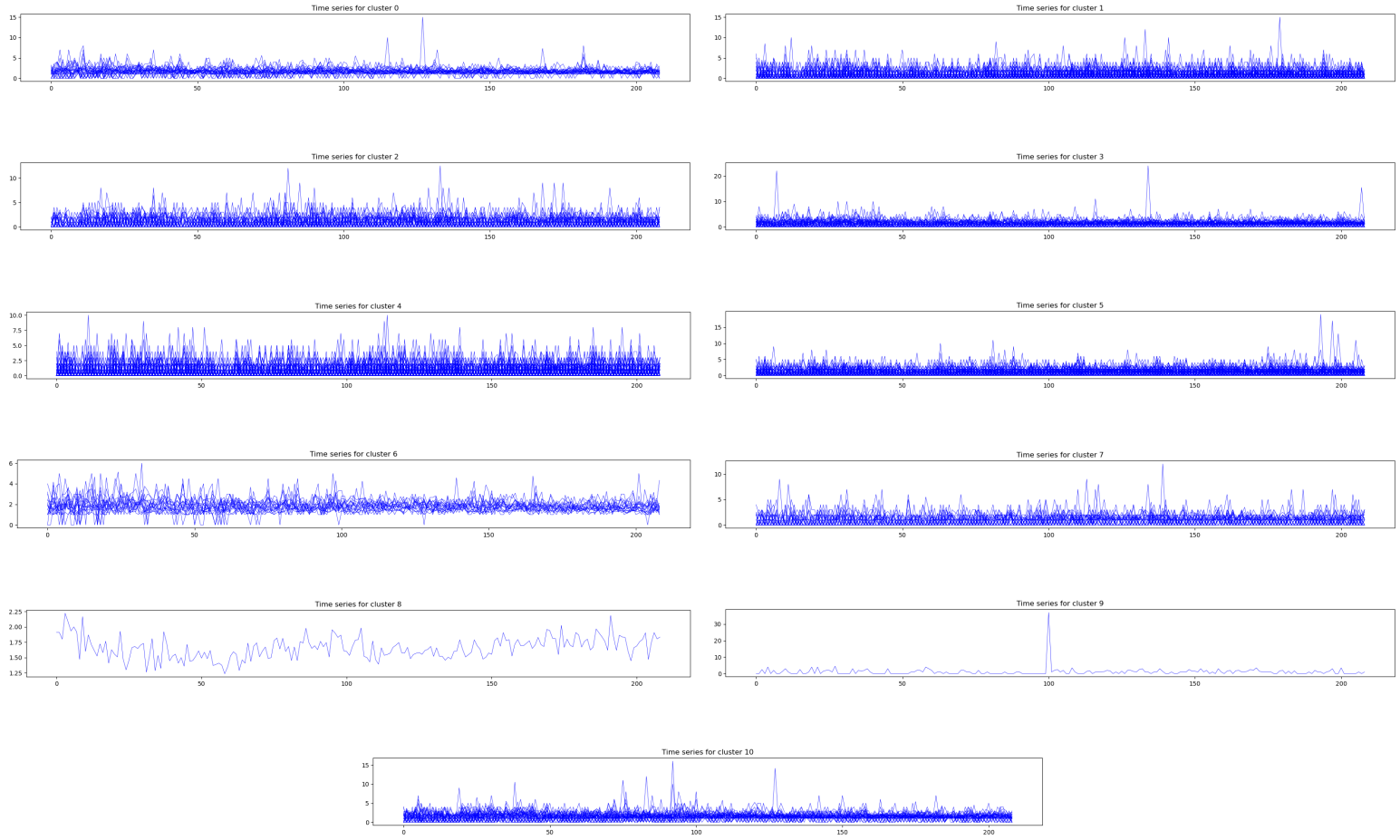
Offset Translation





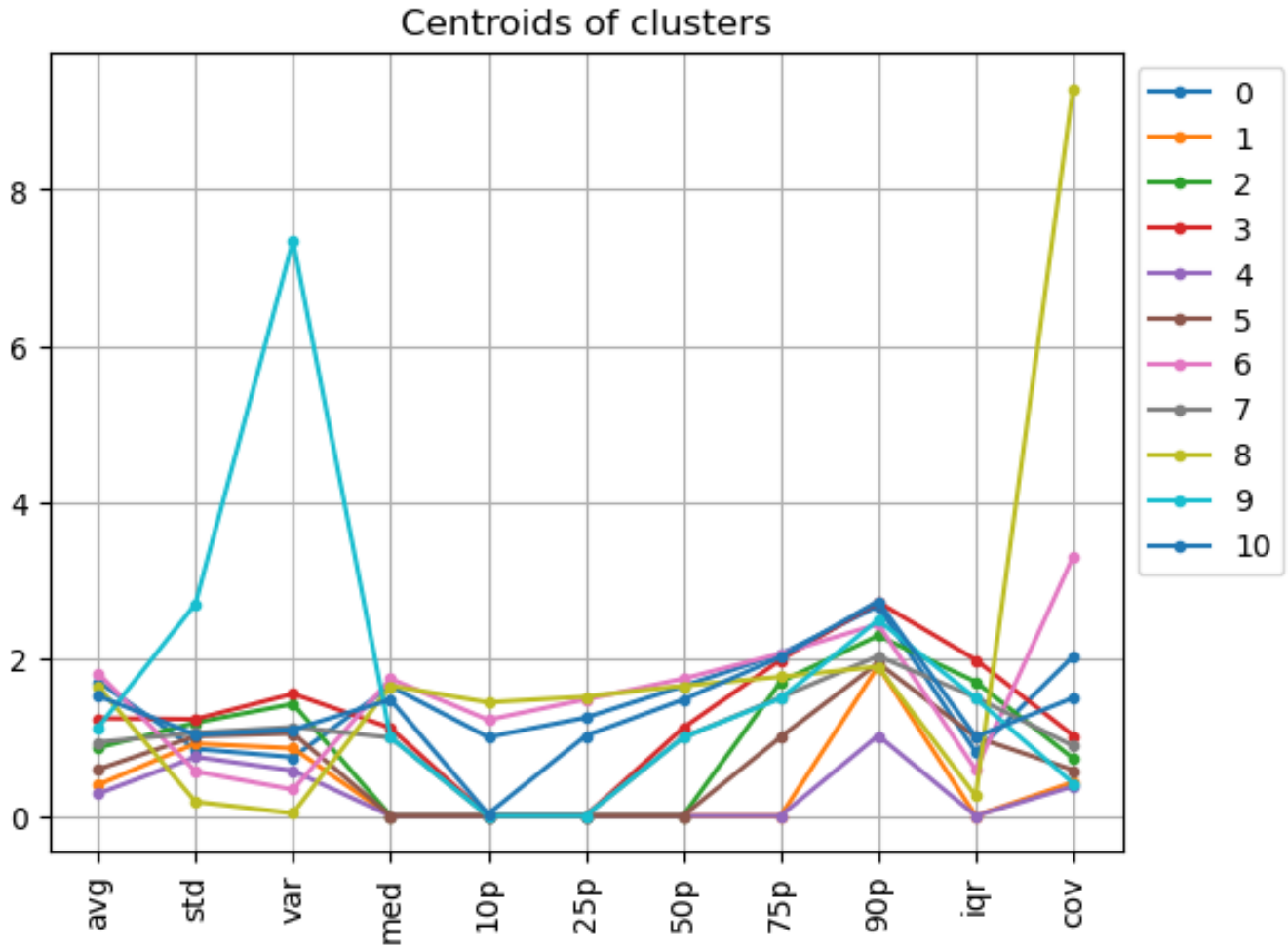
`TimeSeriesScalerMeanVariance` from `tslearn`, ($\mu = 0.0, std = 1.0$). The scaled time series were utilized for feature-based K-Means clustering.

Feature-based K-Means clustering



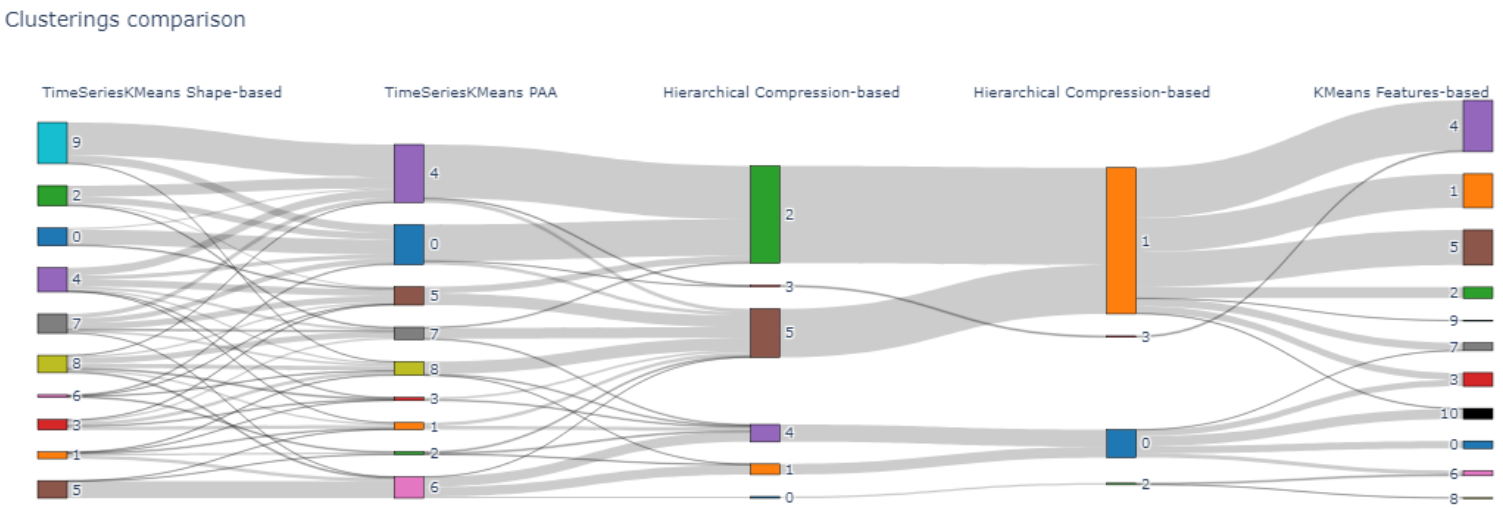
Time series clustering results obtained through the Feature-based K-Means algorithm.

Feature-based K-Means clustering - centroids



Clustering results of time series using the Feature-based K-Means algorithm, the figure highlighting the centroids of each cluster.

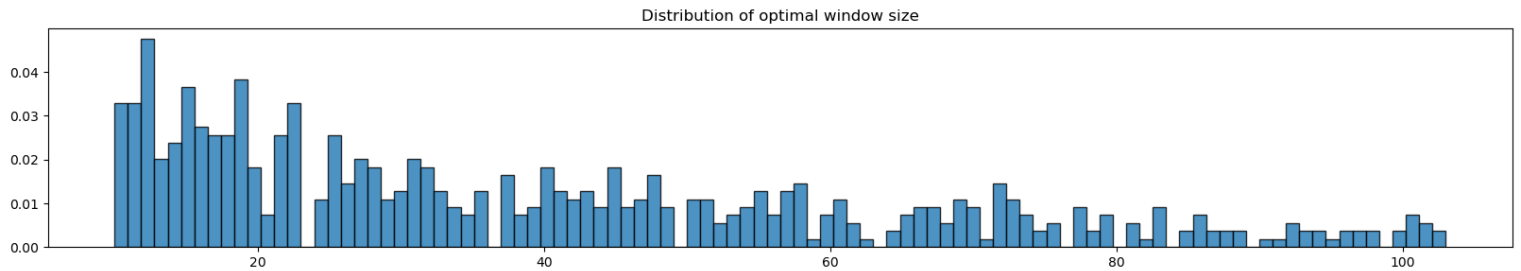
Time series clustering comparison



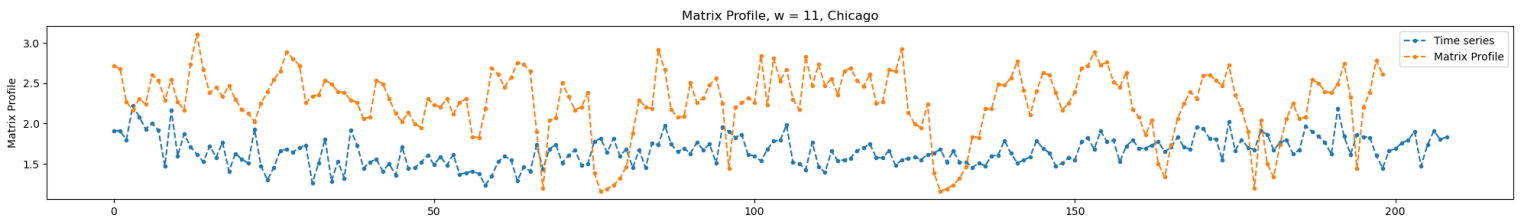
Sankey diagram of the labels assigned by the different time series clustering algorithms.

Matrix profile - optimal window size

The optimal window size to compute the matrix profile was determined for each time series using the Highest Auto-correlation method, as suggested in *"Window Size Selection In Unsupervised Time Series Analytics: A Review and Benchmark*[1] *"*.



The optimal window size was determined for all the series, yielding a mean value of 39.15 with a standard deviation of 24.62. The minimum optimal window size observed was 10, while the maximum reached 103.



[1] Ermshaus et al.

https://project.inria.fr/aaltd22/files/2022/08/AALTD22_paper_3876.pdf

Shaplets classifier comparison

Comparison of Macro F1 Averages Between Labels obtained by Different Classifiers.

	ShapeletModel	DecisionTree	KNN
DecisionTree	0.667	-	-
KNN	0.693	0.717	-
Rocket	0.719	0.617	0.635

- RocketClassifier and ShapletModel classified a higher number of data points similarly, this highlights consistency in predictions between the two.
- Decision Trees and K Nearest Neighbors classifiers show consistency in predictions. This result was expected, as both models utilized the same set of shaplets.