On the selection of unsupervised algorithms for intrusion detection
-
Recent results at the University of Florence

Presenter: **Andrea Ceccarelli**
University of Florence

Work done with: Andrea Bondavalli, Tommaso Capecchi, Lorenzo Salani, Tommaso Zoppi

RCL
RESILIENT COMPUTING LAB

UNIVERSITÀ DEGLI STUDI FIRENZE
DIMAI
DIPARTIMENTO DI MATEMATICA E INFORMATICA "ULISSE DINI"

Presentation Outline

Very few basics on anomalies and their detection

Introduction: recap on results in last year or so

Recent work: selection of algorithms relating attacks to anomaly classes

Conclusions and Future Works

Presentation Outline

Very few basics on anomalies and their detection

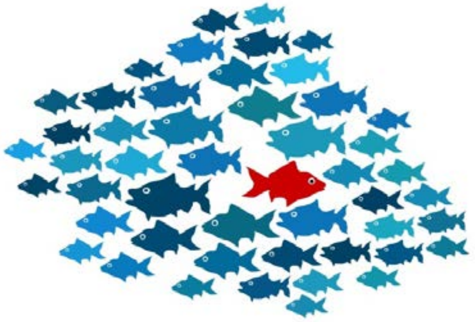
Introduction: recap on results in last year or so

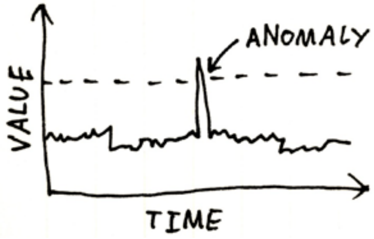
Recent work: selection of algorithms relating attacks to anomaly classes

Conclusions and Future Works

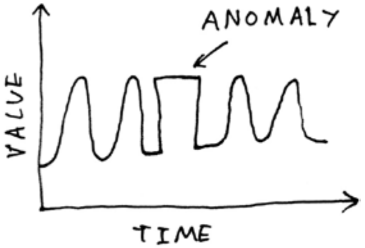
What are anomalies?

Anomaly detection refers to the problem of **finding patterns in data that do not conform to an expected behaviour**

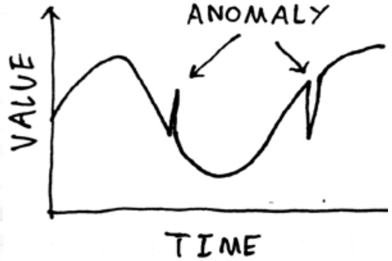




Point: a single data instance that is not compliant with the usual trend of a variable



Collective: a collection of related data instances that is anomalous with respect to the entire data set.



Contextual: a single data instance that is unexpected in a given context;

Anomaly-based Intrusion and Error detection

Anomalies can point to significant information in a wide variety of application domains, e.g.

- **Dependability**: Software bugs, Misconfigurations
- **Security**: **Malware, Attacks ← we focus here**

Unsupervised learning algorithms

- often considered the most suitable to identify unknown errors or zero-day attacks
- no need for labels in the data, despite
  - they help tailoring algorithms' parameters
  - they are required for validation to derive metric scores

Presentation Outline

Very few basics on Anomalies and their Detection

Introduction: recap on results in last year or so

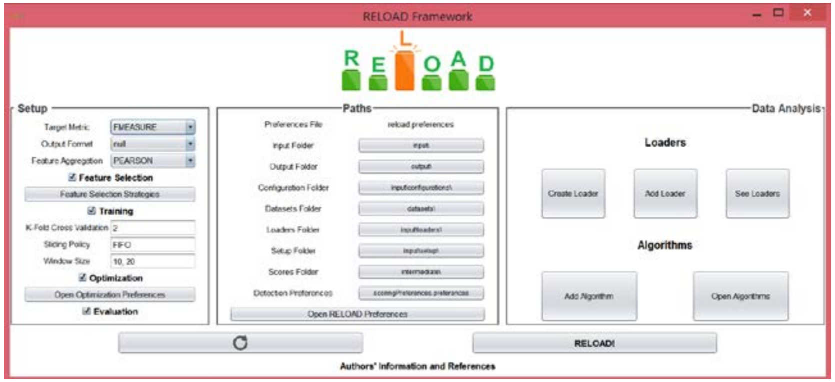
Recent work: selection of algorithms relating attacks to anomaly classes

Conclusions and Future Works

RELOAD: Rapid EvaLuation of Anomaly Detectors

Specifically crafted with attacks and errors datasets in mind

- Includes support for sliding windows
- GUI; we tried to keep it as simple as possible
- Includes 10 features selection strategies, 23 algorithms, 11 metrics

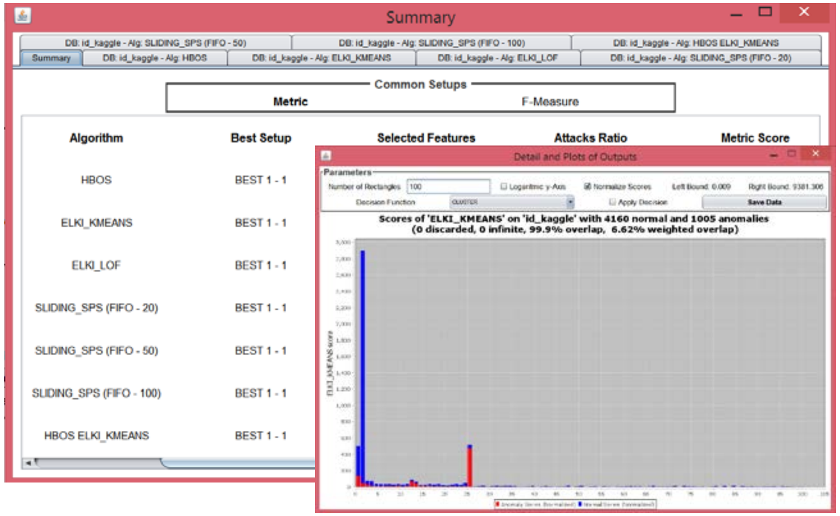


RELOAD Framework

Setup
- Target Metric: FMEASURE
- Output Format: null
- Feature Aggregation: PEARSON
- ☑ Feature Selection
- ☑ Training
- K-Fold Cross Validation: 2
- Sliding Policy: FIFO
- Window Size: 10, 20
- ☑ Optimization
- ☑ Evaluation

Paths
- Preferences File: reload.preferences
- Input Folder: input
- Output Folder: output
- Configuration Folder: input\configurations\
- Datasets Folder: datasets\
- Loaders Folder: input\loaders\
- Setup Folder: input\setup\
- Scores Folder: intermediate\
- Detection Preferences: scoringPreferences.preferences

R E L O A D
Feature Selection Strategies
Open Optimization Preferences
Open RELOAD Preferences
Data Analysis
Loaders
Create Loader
Add Loader
See Loaders
Algorithms
Add Algorithm
Open Algorithms
RELOAD!
Authors' Information and References



Summary

DB: id_kaggle - Alg: SLIDING_SPS (FIFO - 50)
DB: id_kaggle - Alg: SLIDING_SPS (FIFO - 100)
DB: id_kaggle - Alg: HBOS ELKI_KMEANS
Summary
DB: id_kaggle - Alg: HBOS
DB: id_kaggle - Alg: ELKI_KMEANS
DB: id_kaggle - Alg: ELKI_LOF
DB: id_kaggle - Alg: SLIDING_SPS (FIFO - 20)

Common Setups

Metric	F-Measure

Algorithm	Best Setup	Selected Features	Attacks Ratio	Metric Score
HBOS	BEST 1 - 1			
ELKI_KMEANS	BEST 1 - 1			
ELKI_LOF	BEST 1 - 1			
SLIDING_SPS (FIFO - 20)	BEST 1 - 1			
SLIDING_SPS (FIFO - 50)	BEST 1 - 1			
SLIDING_SPS (FIFO - 100)	BEST 1 - 1			
HBOS ELKI_KMEANS	BEST 1 - 1			

Detail and Plots of Outputs

Parameters
- Number of Rectangles: 100
- ☐ Logaritmic y-Axis
- ☑ Normalize Scores
- Left Bound: 0.009
- Right Bound: 9381.306
- Decision Function: CLUSTER
- ☐ Apply Decision

Scores of 'ELKI_KMEANS' on 'id_kaggle' with 4160 normal and 1005 anomalies
(0 discarded, 0 infinite, 99.9% overlap, 6.62% weighted overlap)

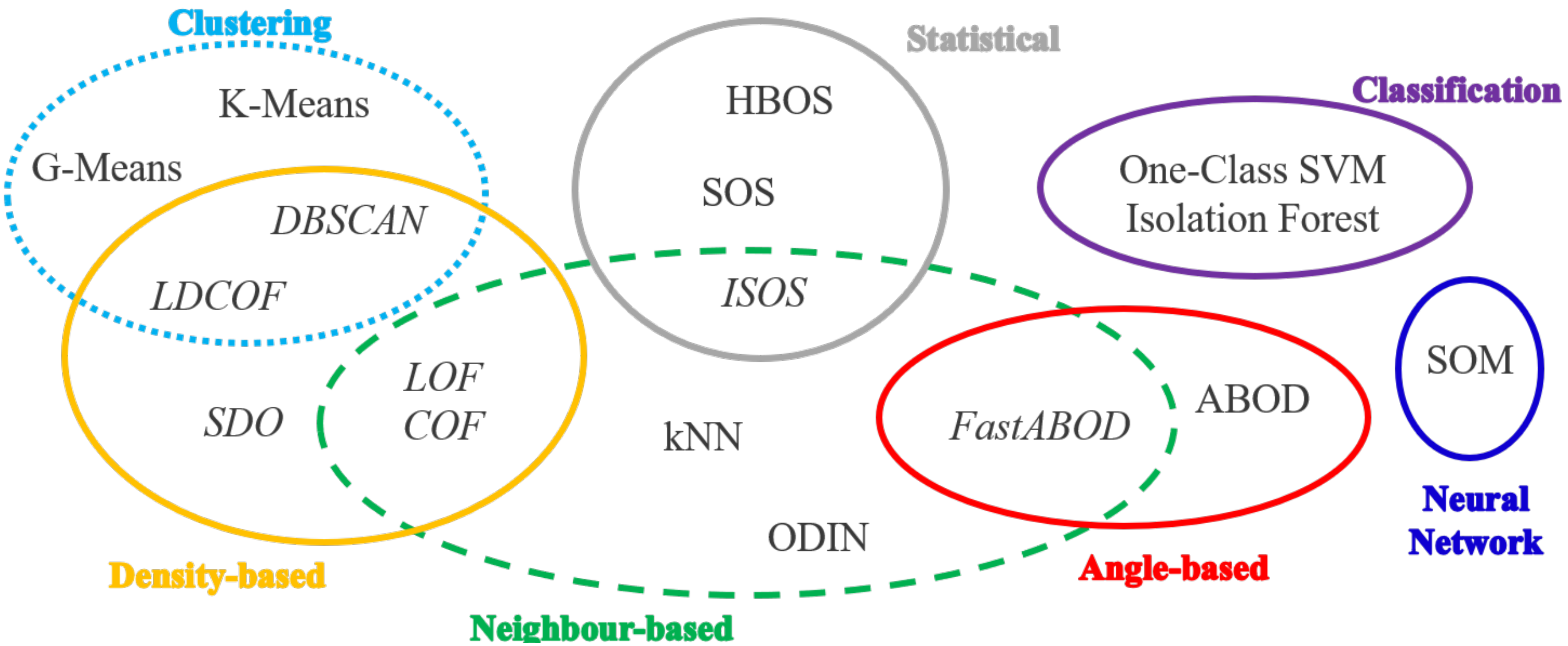
Zoppi, T., Ceccarelli, A., Bondavalli, A. Evaluation of Anomaly Detectors Made Easy with RELOAD. ISSRE 2019 (Tool paper)

Downloadable at (GPL3 license): https://github.com/tommyippoz/RELOAD

Extensive comparison of algorithms

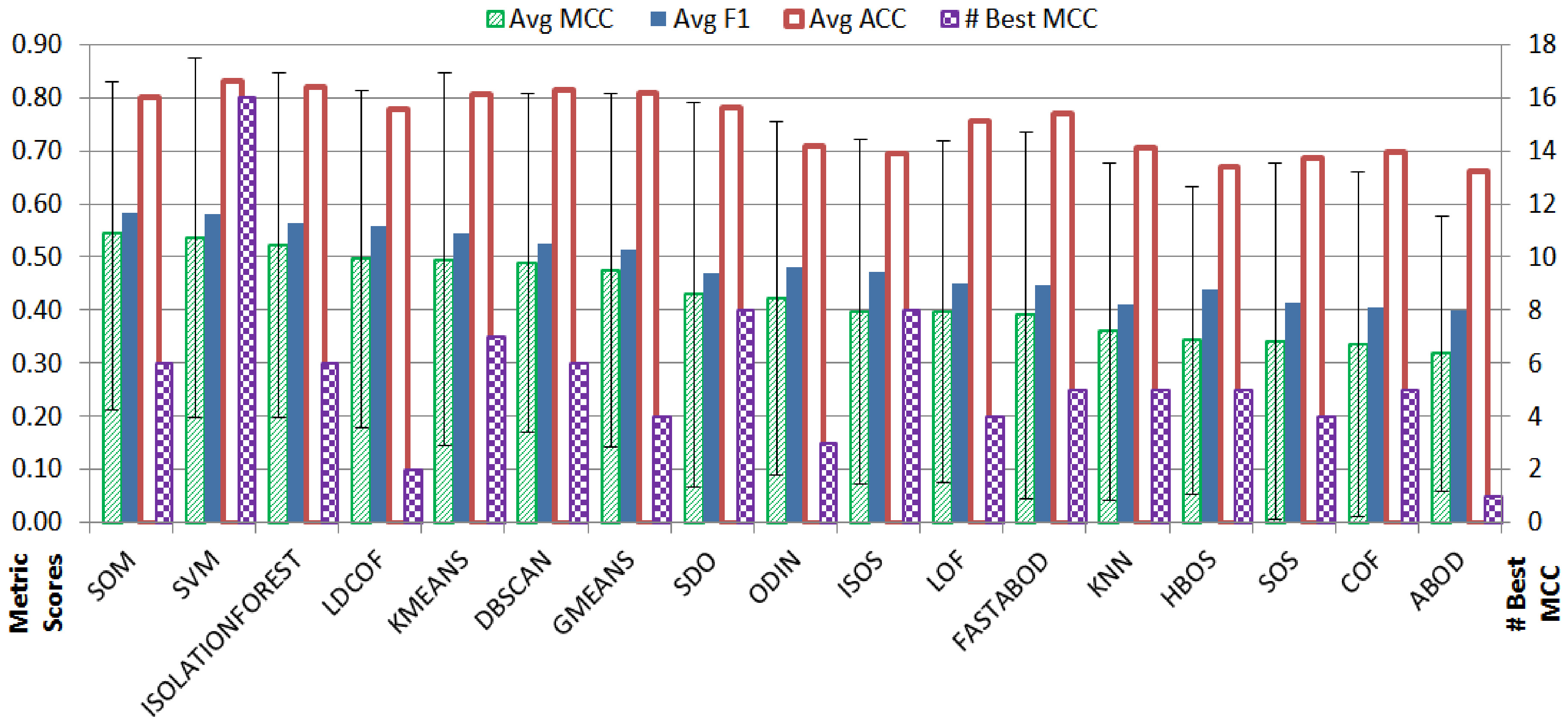
RELOAD exploited to investigate

- 17 algorithms belonging to the main families
- using 11 attacks datasets



(a first version of this study appeared at ACM SAC 2019)

A sample of results



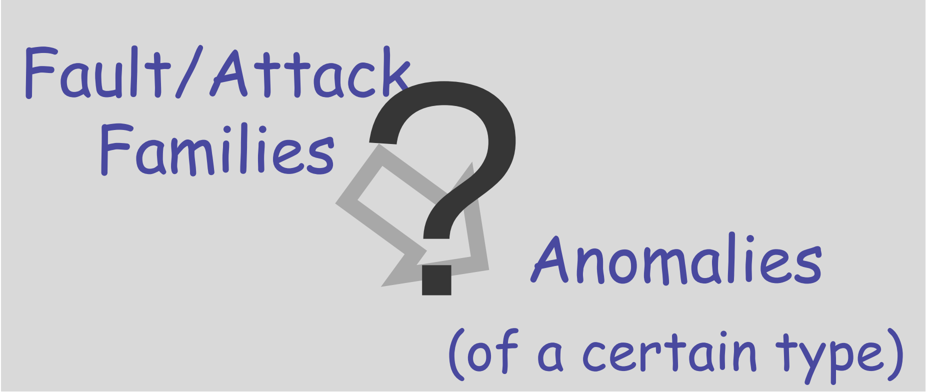
MCC as reference metric - good also for unbalanced datasets.

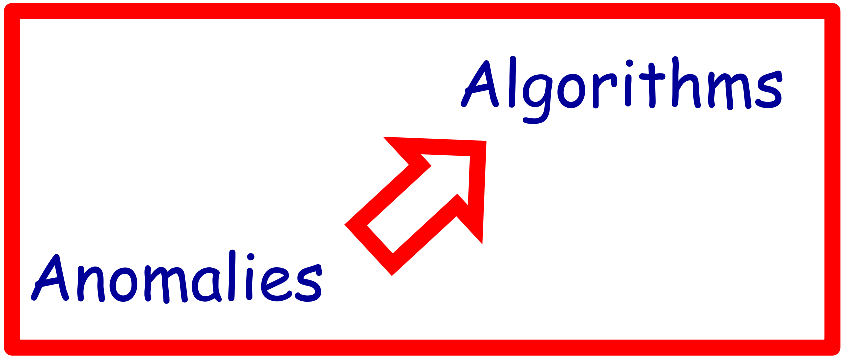
$$\mathrm{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

On detectability of anomaly types

Suitability of unsupervised algorithms to detect point, contextual or collective anomalies

- non-sliding algorithms are capable to detect point and collective anomalies
- sliding window algorithms are better with contextual anomalies





T. Zoppi, A. Ceccarelli, and A, Bondavalli. "On Algorithms Selection for Unsupervised Anomaly Detection." *PRDC 2018*

Presentation Outline

Very few basics on Anomalies and their Detection

Introduction: recap on results in last year or so

Recent work: selection of algorithms relating attacks to anomaly classes

Conclusions and Future Works

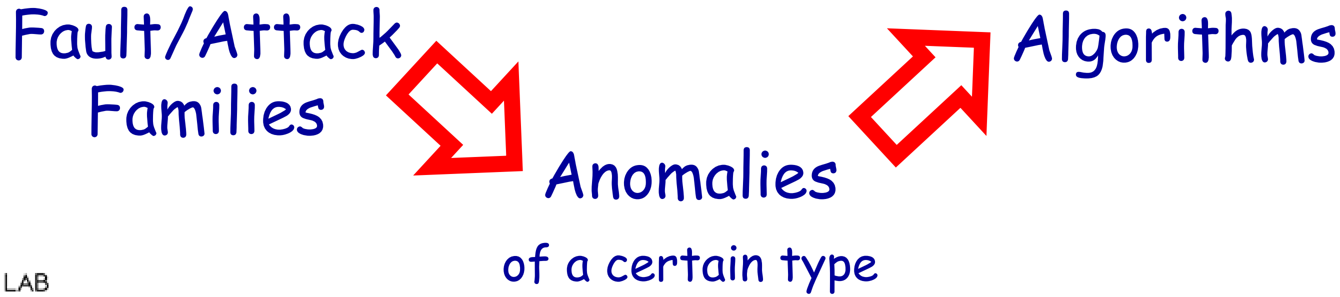
Our objective (refined from last slide)

How to select algorithm(s) that maximizes detection capability?

- We study relations between attack families, anomaly classes and algorithms

Implications:

- an unknown attack belonging to an attack family is most likely to get observed by unsupervised algorithms that are particularly effective on such attack family.
- Consequently, rules can be defined to select algorithms based on "target" attack families


Fault/Attack Families

Anomalies

of a certain type

Algorithms

Brief review of the approach: main steps

1. Algorithms Selection, from the different families
2. Identification of labelled datasets
3. Attack families
4. Metrics identification
5. Attack and data inspection, experimental analysis, results ← rest of the talk

Attack Family	Mapping of Attacks
Communication - Passive	(KDD99, NSL-KDD09) Probing, (ISCX12) Infiltration, (UNSW-NB15) Reconnaissance, Analysis
Communication - Active	(ISCX12) Bruteforce, DDoS, (KDD99, NSL-KDD09) DoS, (UNSW-NB15) Fuzzers, Backdoor
Host	(KDD99, NSL-KDD09) U2R, R2L, (UNSW-NB15) Worms, Shellcode
Application	(UNSW-NB15) Exploits, Generic

UNIVERSITÀ DEGLI STUDI FIRENZE
DIMAI DIPARTIMENTO DI MATEMATICA E INFORMATICA "ULISSE DINI"

Match attacks to anomaly classes

First, associate each attack in the datasets to an anomaly type, based on

i) Attacks description and

ii) manual inspections (plus some statistics)

Then, we confirm with experiments

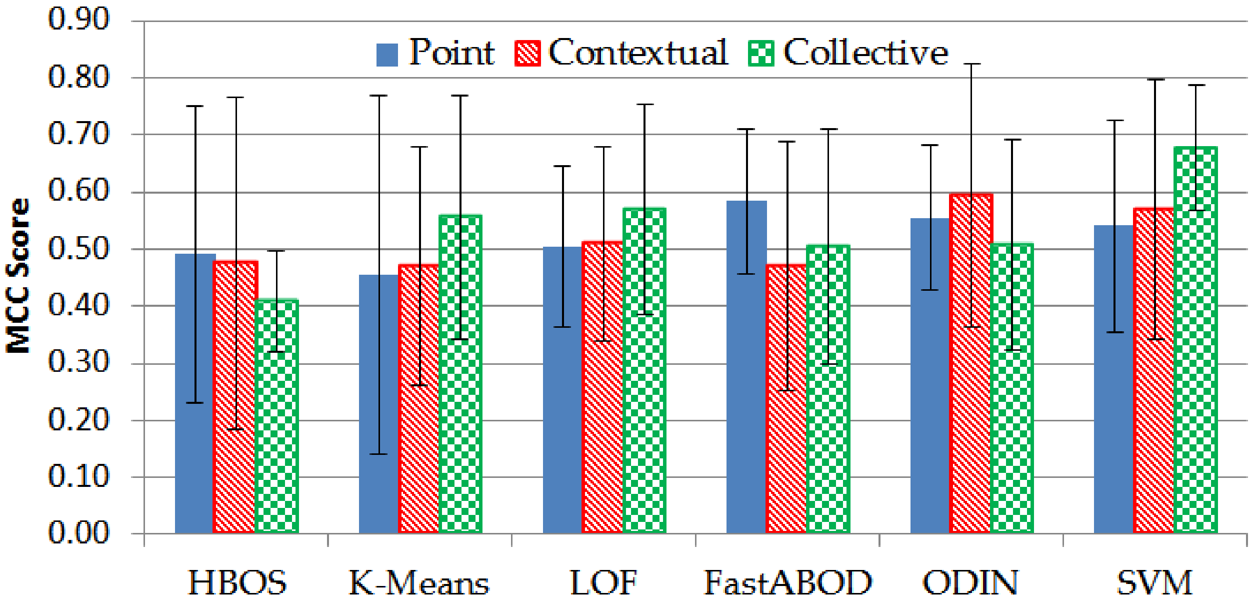
- Train the algorithms to detect a specific attack
- Use other attacks for evaluation. Should be more effective on attacks of the same category.

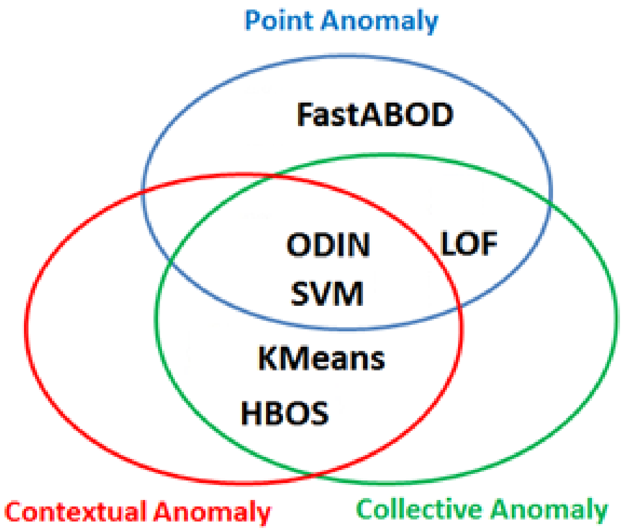
Dataset	Training		Testing		MCC
UNSW-NB15 2015	**Worms (Host type)**	**Contextual**	**Worms**	**Contextual**	**0.74**
			Shellcode (Host type)	Contextual	0.64
			Exploits (App. type)	Point	0.47
			Reconnaissance (Comm.type)	Collective	0.38

Match algorithms to anomaly classes

We proceed in two steps:

- First, we run algorithms on synthetic datasets in which collective, contextual and point anomaly are introduced
- Then, we execute on real datasets





An extract of concluding results

Attack Families	Anomaly Classes	Algorithms (Families)	Motivation
Host (e.g., Worm, Shellcode)	Contextual	ODIN (Neighbour), SVM (Classification)	ODIN slightly better than SVM for contextual. If also point anomalies are expected, SVM is to be preferred.
		…	
Balanced Coverage	Point, Contextual, Collective	ODIN (Neighbour), SVM (Classification)	Both algorithms show overall balanced capabilities in identifying all anomaly classes, without exposing evident vulnerabilities.

Presentation Outline

Very few basics on Anomalies and their Detection

Introduction: recap on results in last year or so

Recent work: selection of algorithms relating attacks to anomaly classes

Conclusions and Future Works

Conclusions and Future Works

Presentation of main activities we did on the topic in recent times.

- Latest focus is on the definition of guidelines that can be used when setting up an anomaly detection systems

Next steps (just started):

- Combined usage of different algorithms: ability to distinguish the type of anomaly at runtime and trust algorithms depending on the kind of anomaly
- Feature selections strategies and dataset complexity measures as indication of performances of algorithms

Q&A Time

THANK
YOU
GRACIAS
ARIGATO
SHUKURIA
DANKSCHEEN
TASHAKKUR ATU
YAQHANYELAY
SUKSAMA
BİYAN
SHUKRIA
GRAZIE
MEHRBANI
JUSPAXAR
BOLZİN
MERCI

Q
&
A