Detecting organized eCommerce fraud using scalable categorical clustering

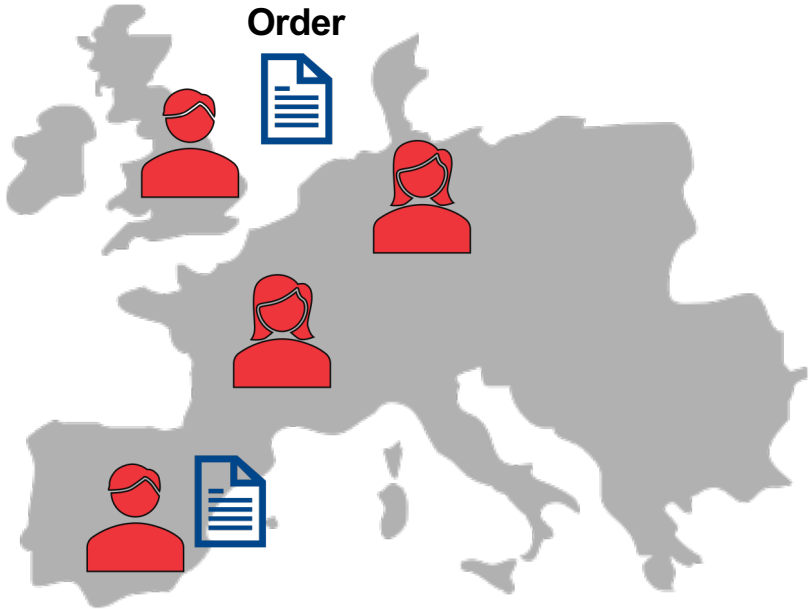
ACSAC'19

Aalto-yliopisto
Aalto-universitetet
Aalto University

Samuel Marchal, *Aalto University & F-Secure Corporation*
Sebastian Szyller, *Aalto University*

Contact: samuel.marchal@aalto.fi

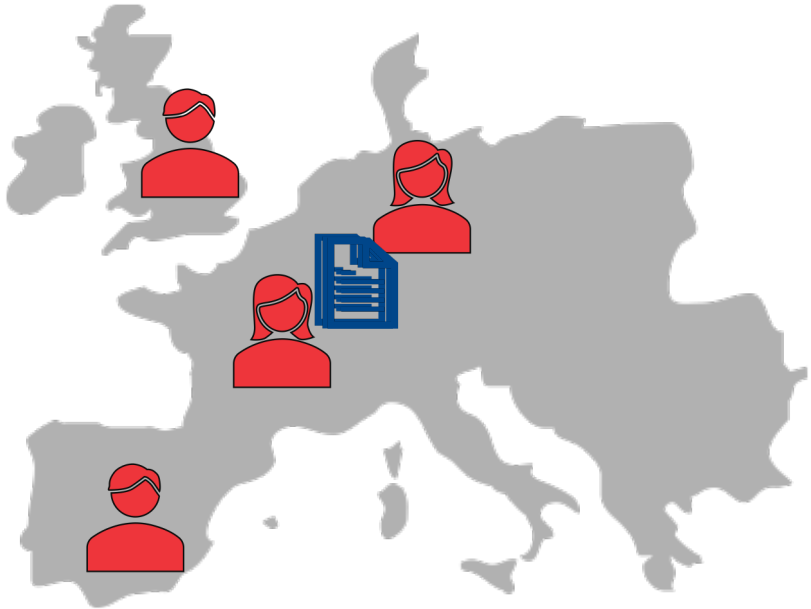
eCommerce fraud



Aalto-yliopisto
Aalto-universitetet
Aalto University

3-5% of orders is fraudulent for $50B loss every year

Organized eCommerce fraud



Fraud detection system

amazon

zalando

Alibaba

Observation and assumption

Organized frauds belong to fraud campaigns characterized by

- Small number of customers (fraudsters)
- Several orders (10s)
- Limited time period (several days)
- Close geographical location (same city / neighborhood)
- Similar payment method (credit card / delayed bills)

By grouping orders into fraud campaigns
we can improve the detection of organized fraud

→ Orders grouped together are likely fraud
→ Isolated orders are likely legitimate

Clustering organized fraud

Cluster frauds from the same fraud campaign together

- Generate small clusters → 1 cluster = 1 fraud campaign = 10s of orders
- Generate singletons → legitimate order can remain non-clustered

Process 100,000s orders in a few hours

- Scalable method with low computational complexity

Input categorical data

- Fraud campaign similarity = identical categorical attributes
  → email address, pickup point, street address, city, credit card number, etc.

Categorical clustering

Existing solutions

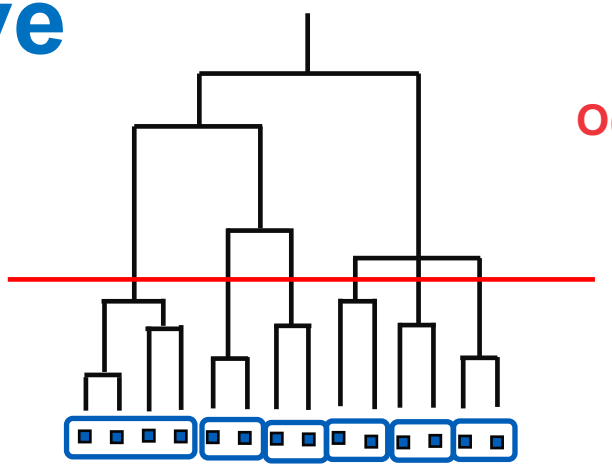
- Can generate small clusters, e.g., agglomerative clustering…
  … but requires pairwise distance computation $O(n^2)$ → not scalable
- Can have acceptable complexity (independent from dataset size), e.g., Kmodes…
  …but designed to generate only large clusters + no singletons

Require a new categorical clustering solution scalable and that can generate small clusters

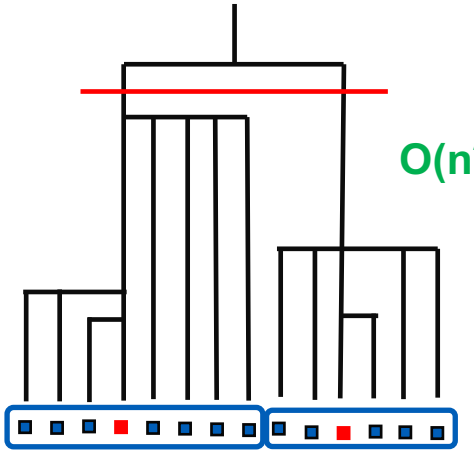
Recursive Agglomerative Clustering Principle

Combine 2 clustering methods

- Agglomerative clustering: ***AggloClust***
  → generate small clusters
  → high complexity
- Clustering with sampling: ***SampleClust***
  → complexity reduction
  → generate large clusters (at most |sample|)



$O(n^2)$



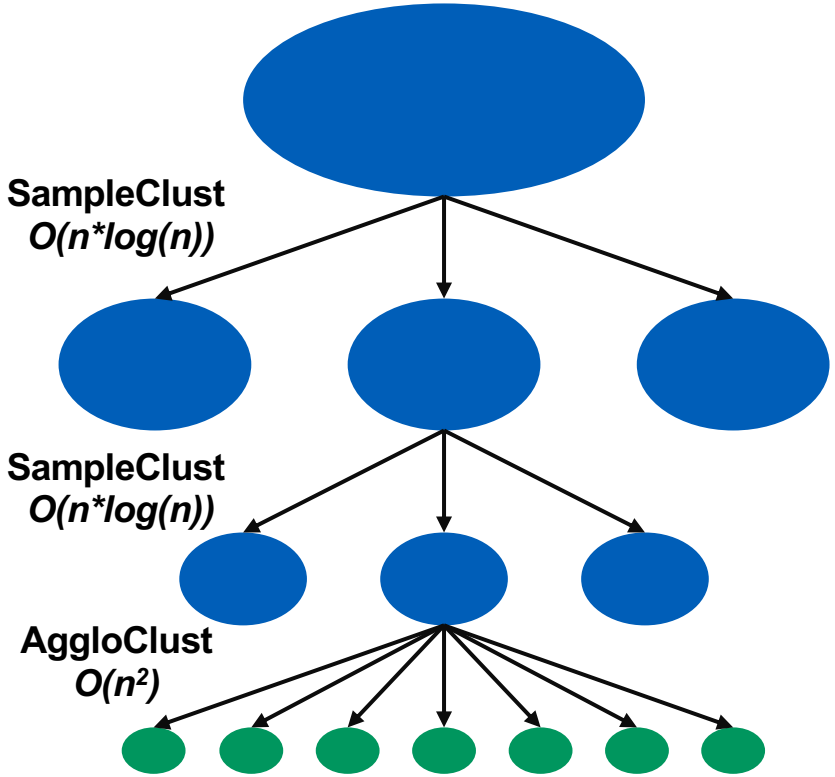
$O(n*|sample|)$

RecAgglo Algorithm

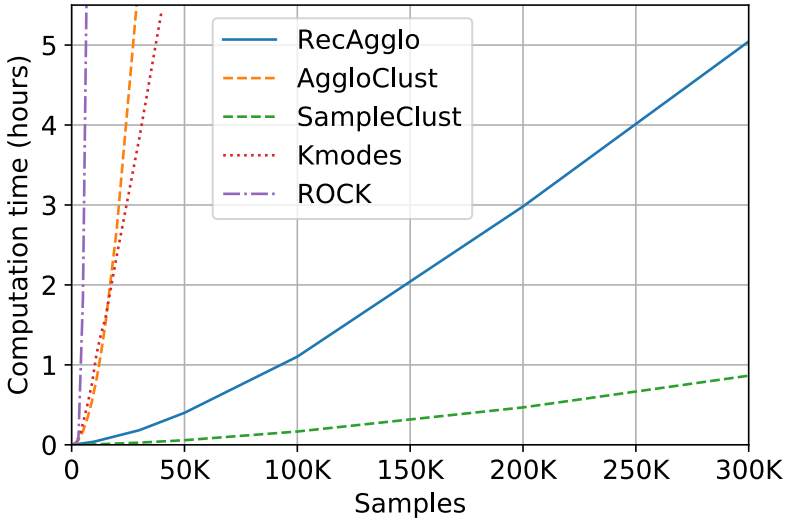
Selectively apply *AggloClust* and *SampleClust*

1. Recursively split clusters using ***SampleClust***
   - Until they are “small enough”
   - Random sample of size |sample| = log(n)

2. Finalize small clusters using ***AggloClust***
   - High quality small clusters
   - Possible singletons



RecAgglo speed and accuracy



Clustering 10,000 legitimate orders and 5,000 frauds

Algorithm	Impurity	Clustered fraud	Time
RecAgglo	**0.8%**	**42.1%**	**185s**
AggloClust	1.2%	51.9%	1h31
SampleClust	3.3%	2.2%	38s
Kmodes	10.5%	39.2%	7h44
ROCK	0.9%	30.3%	1h38

Only RecAgglo and SampleClust scale to large datasets

→ 5 hours to cluster 300,000 samples

Only RecAgglo and AggloClust have high detection capability while achieving low impurity

→ potential to detect 42.1% frauds with 0.8% false positives

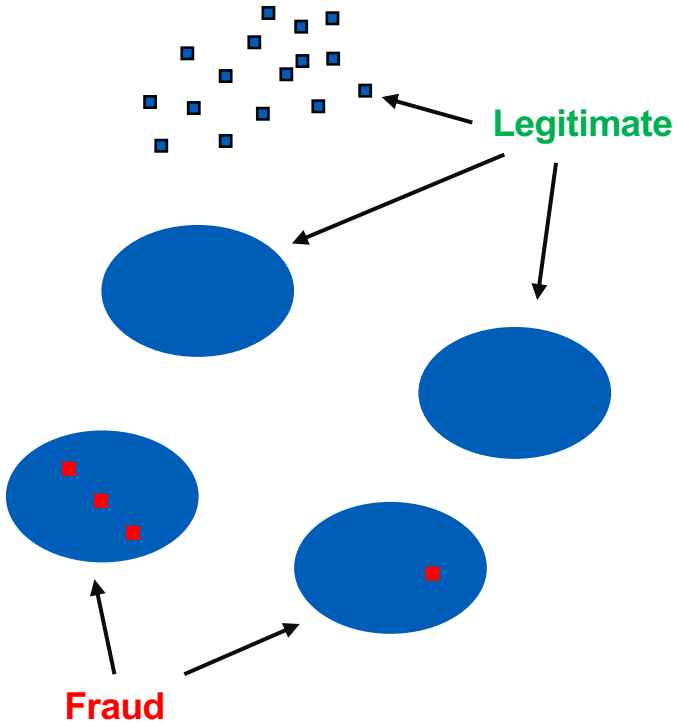
Aalto-yliopisto
Aalto-universitetet
Aalto University

Automated fraud detection

Singletons are legitimate

Clusters are suspicious

- Augment fresh unlabeled orders with older labelled fraud prior to clustering
- Detect as fraud all orders in a cluster that include at least one labelled fraud



Fraud detection accuracy

Datasets

- 6M real orders from Zalando online retailer
- 5 countries
- 35 days

Detection rate (clustered)	Detection rate (all fraud)	Precision	False positive rate
62.6 %	26.4%	35.3%	0.1%

False positives

- not “fraud”…
- … but 94.7% are returned, canceled or partly unpaid orders

Summary

Novel clustering algorithm for categorical data: *RecAgglo*

- Scalable
- Generates small clusters
- Designed for grouping organized fraud

Assessment on 6M orders from top European online apparel retailer: Zalando

- Detect 26.4% of fraud
- Generate 0.1% false alarms
- 95% of false alarms are problematic