APTIMA®
Human-Centered Engineering

Adaptive Data Collection and Archiving Plans for Large-scale Cyber Networks

CLSAC

Session 3: Applications 2

Georgiy Levchuk

31 Oct 2018

Outline

- Challenges in processing cyber data
  - "Behavior"-based analytics
- Planning **collection** and **retention** as methods to *scale up* processing
- **Energy/variational models** as a general framework for **scalable adaptive data management**

Highlights

- Cyber analytics:
  - Map **normal** cyber-space
  - Detect **attacks**
  - Identify **anomalies**

- Types of reasoning:
  - **Feature-**based
  - Models from **users** (rules) or **machine learning**
  - Reason about **context**

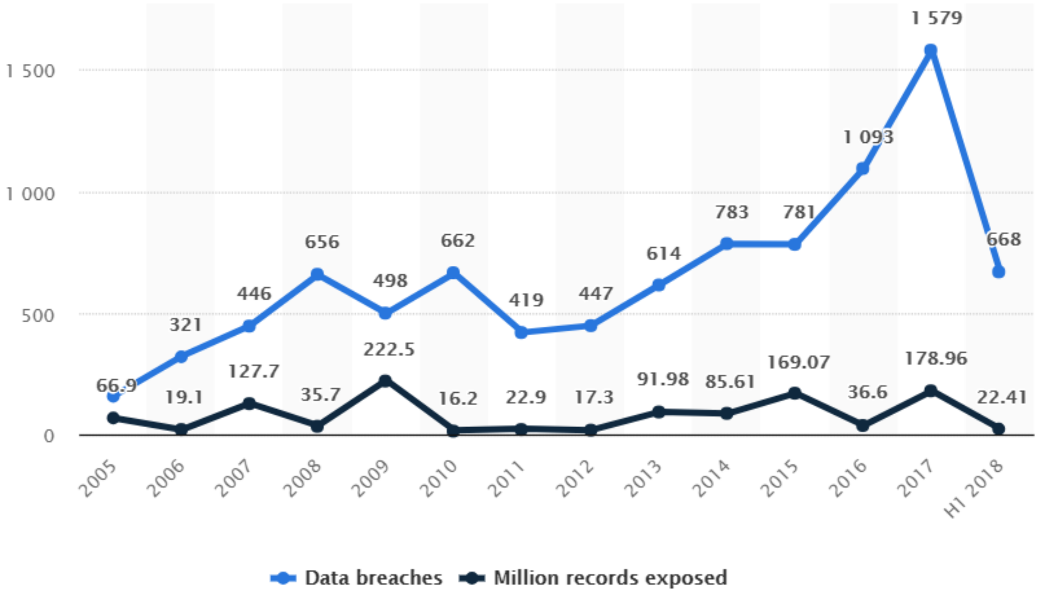
Challenges:

- Data is large
- Training is sparse
- Attacks & environment change

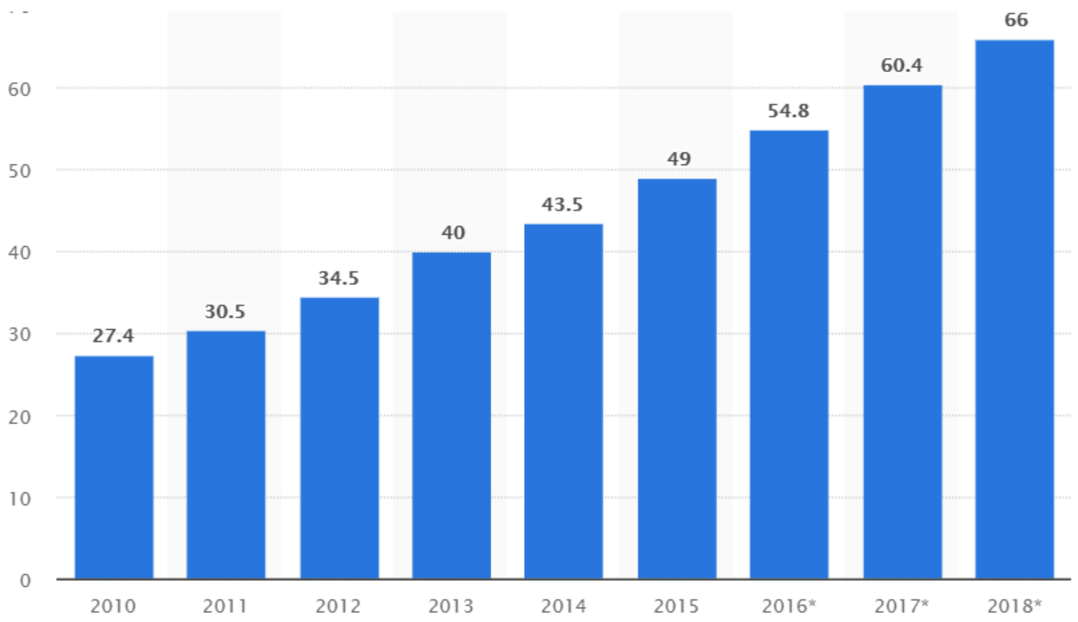
Challenges of scale

- # of cyber-security risks is increasing

- Spending on cyber-security is lagging behind



 # data breaches in the US



Spending in B$

Challenges of scale

- # of cyber-security risks is increasing
- Spending on cyber-security is lagging behind
- Amount of data collected is also growing very rapidly, and cannot be sustained
  - % of data analyzed is shrinking

Amount of security data: **5PB**

end-points: 175M

attack sensors: 126M

threat events / sec: 1K

emails/day: 2.4B

products: 79K

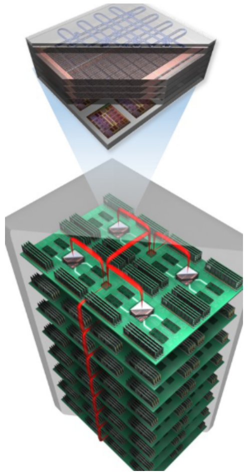
vendors: 25K

Symantec threat collection capabilities

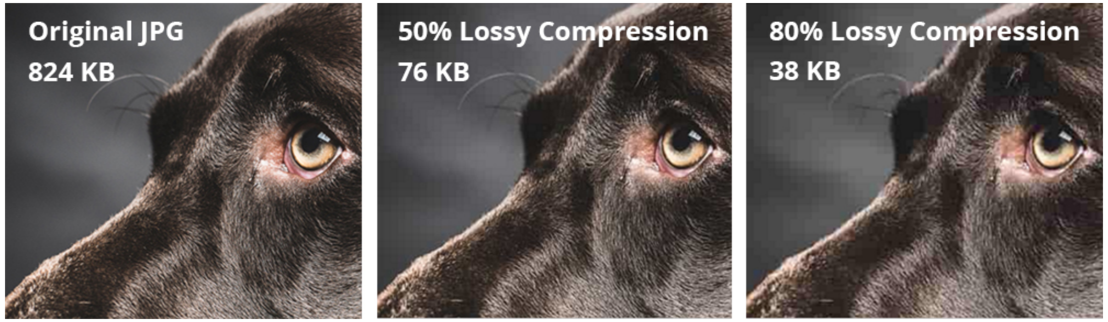
How to scale-up cyber analytics

- More/better compute resources
- Scalable algorithms
  - Better-than-linear complexity
- Data aggregation / compression
- Data sampling & filtering
  - Collection
  - Retention

Large-scale HPC/ data centers



New chips/ electronics



Data compression

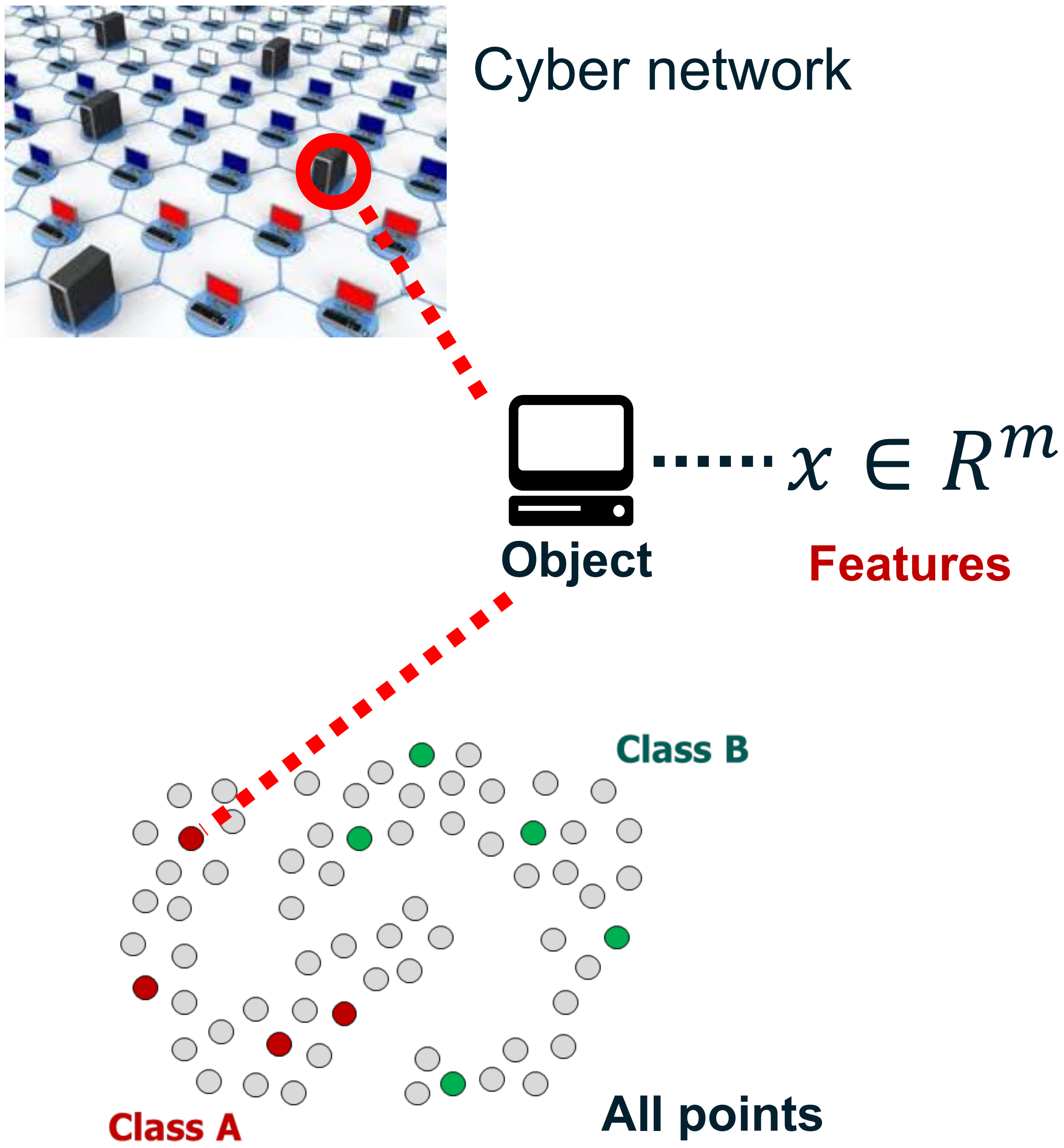
Problems solved by Cyber Analytics

- Formal problems types:
  - Ranking/anomaly detection
  - Node classification/labeling
  - Group detection
  - Joint contextual inference
  - POL learning

- Representative use-cases:
  - Activity classification
  - Botnet detection
  - Stepping-stone attacks
  - Malicious web traffic/attacks

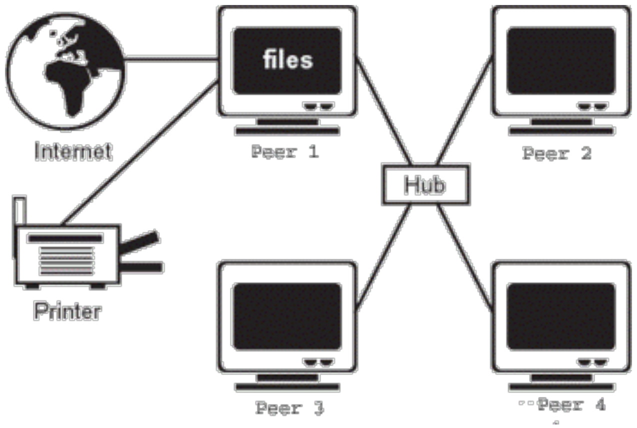
Abstracting cyber activity analysis

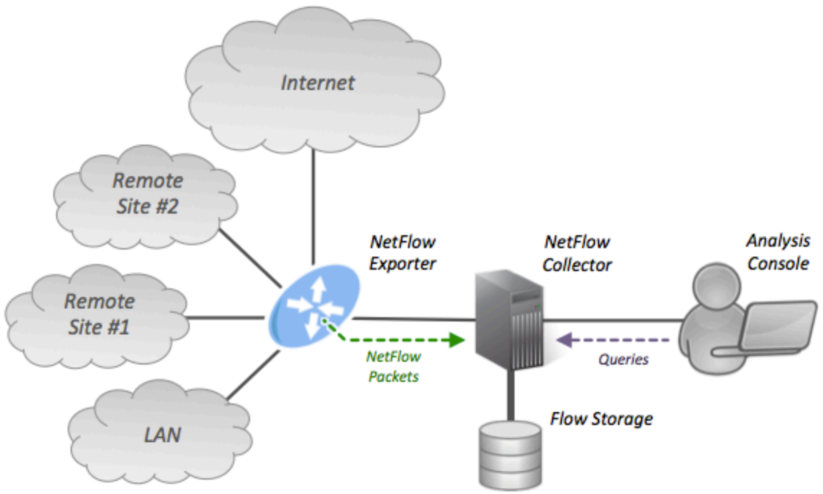
- Cyber data (raw):
  - Host (e.g., event/process log)
  - Network (e.g., flows)
- Objects of analysis:
  - User, IP, (sub)network, organization
- Features:
  - Behavior-based
    - Social, functional, application
  - Event-based
    - IDS, rule-based alerts
  - ML-based

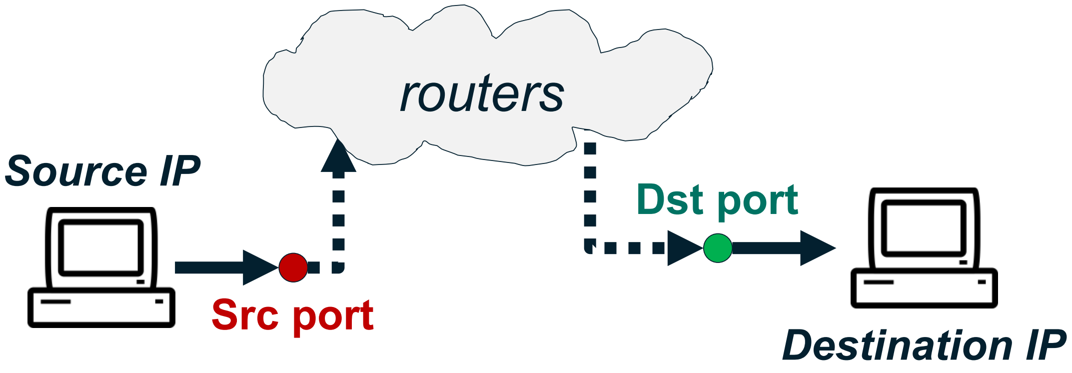


Cyber flow data

- **Social** information:
  - who talks to whom
- **Functional** information:
  - What applications / services are running on the machine (and use which ports)
- Collected at the edge or on local networks

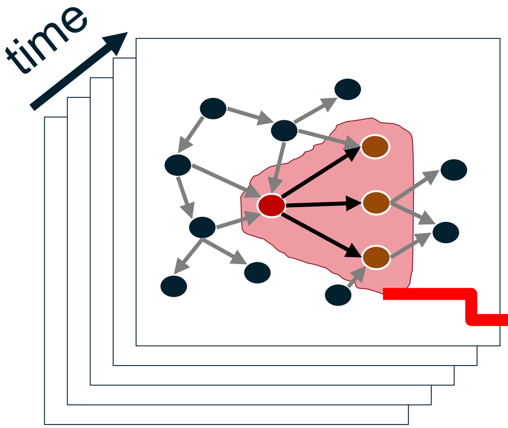






Example "behavioral" features

- Network-based flows can be analyzed to extract social, functional (application), and transport-level information via **application graphs**

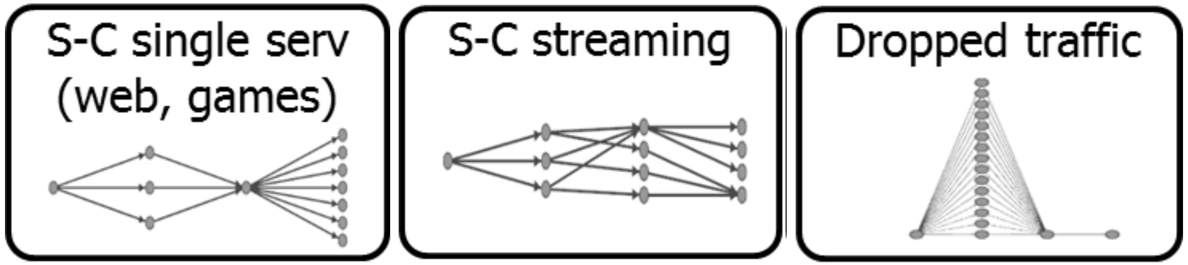


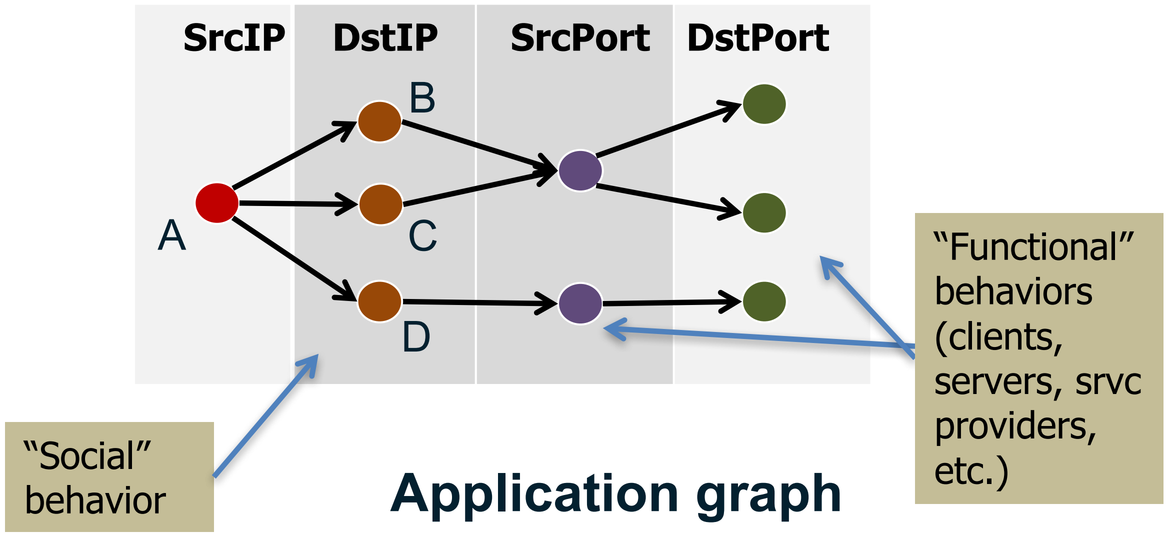
Network X time

Raw NetFlow records

RecID	Src IP	Src Port	Dst IP	Dst Port
1	A	23	B	5433
2	A	23	C	6711
3	A	23	C	5433
4	A	80	D	877

- Features are obtained using topological application graph patterns
  - E.g.:

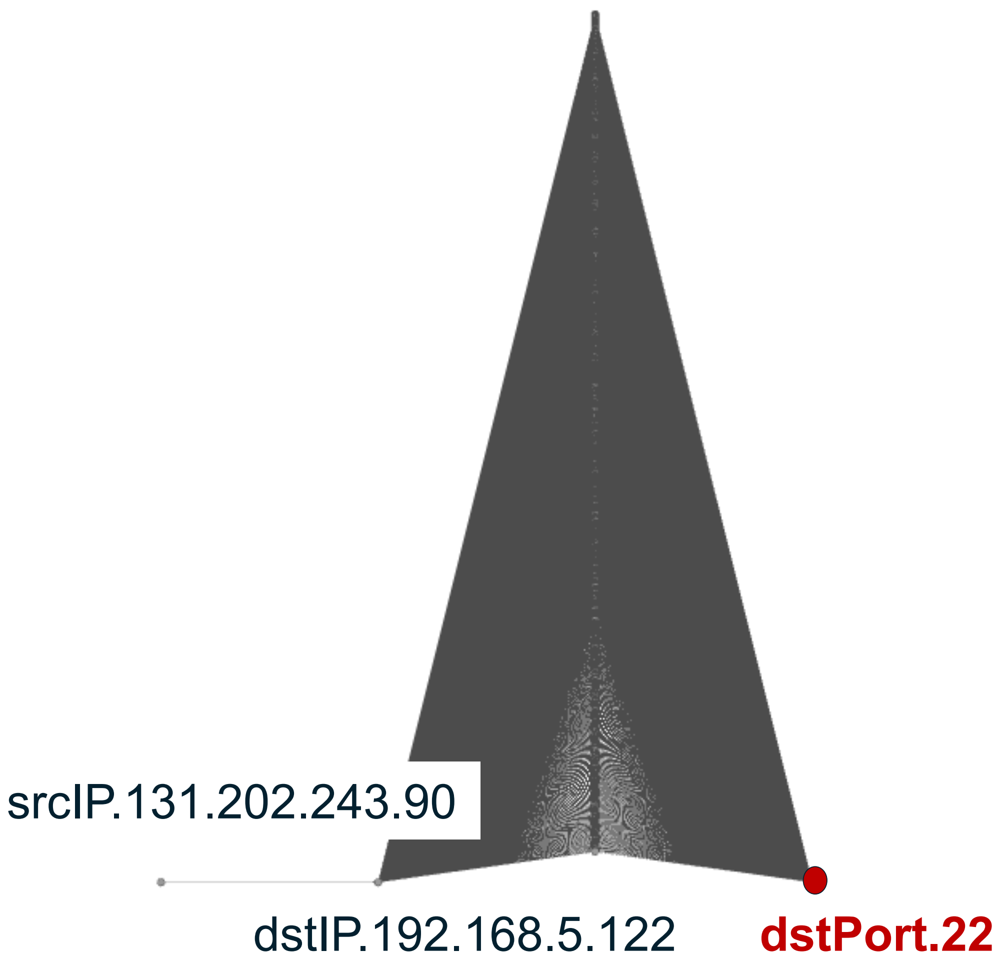




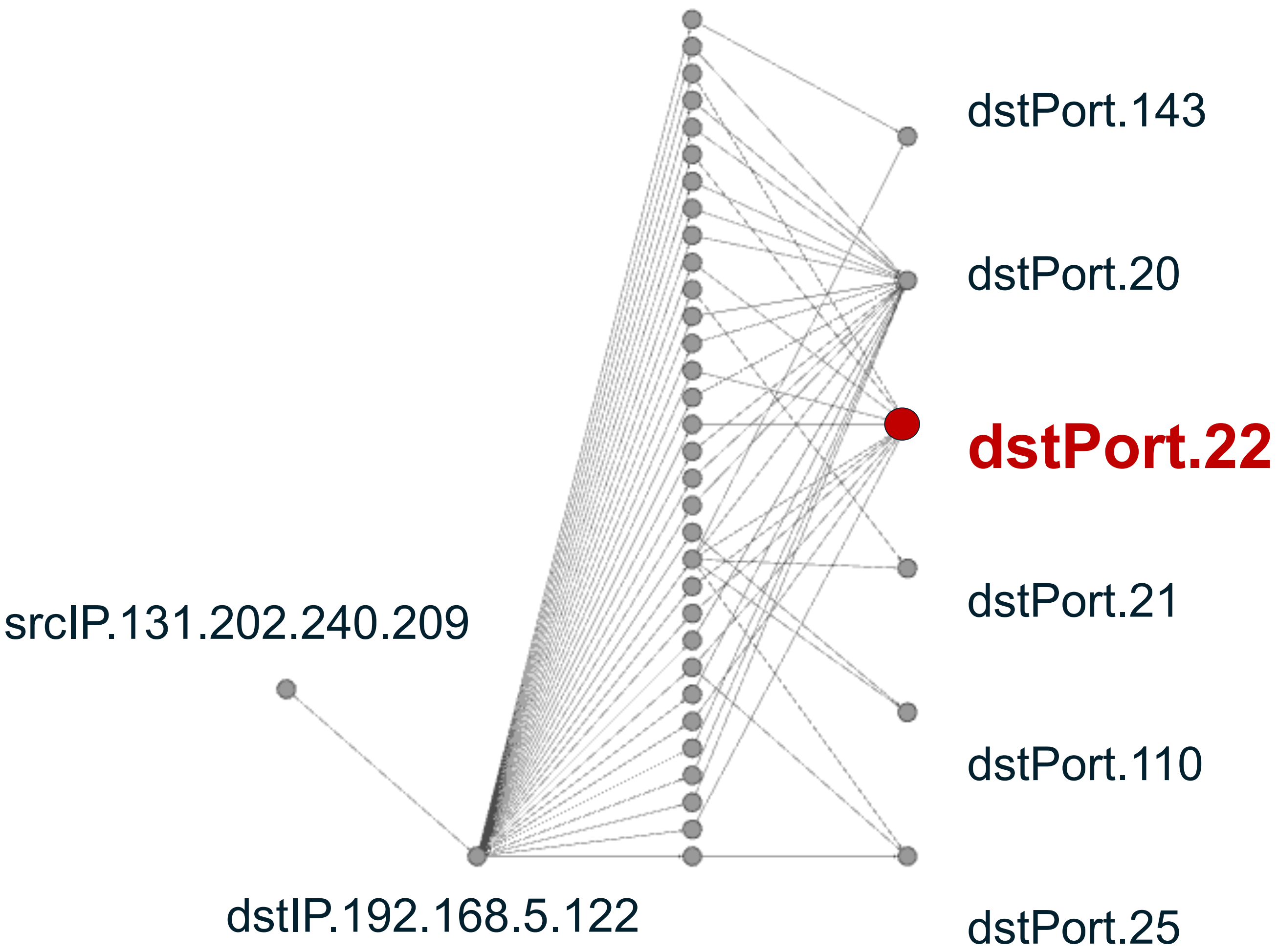
Application graph

Disambiguation power: Attack vs Normal

Attack SSH

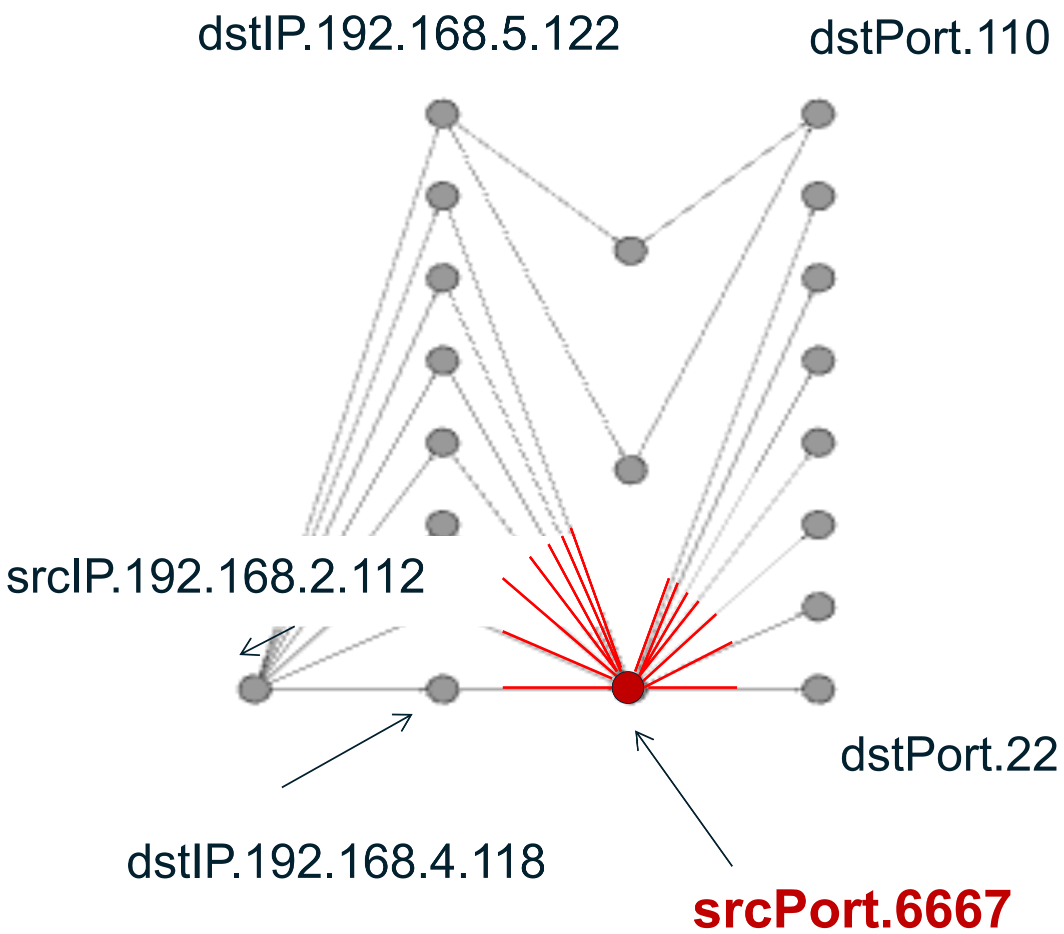


Normal SSH

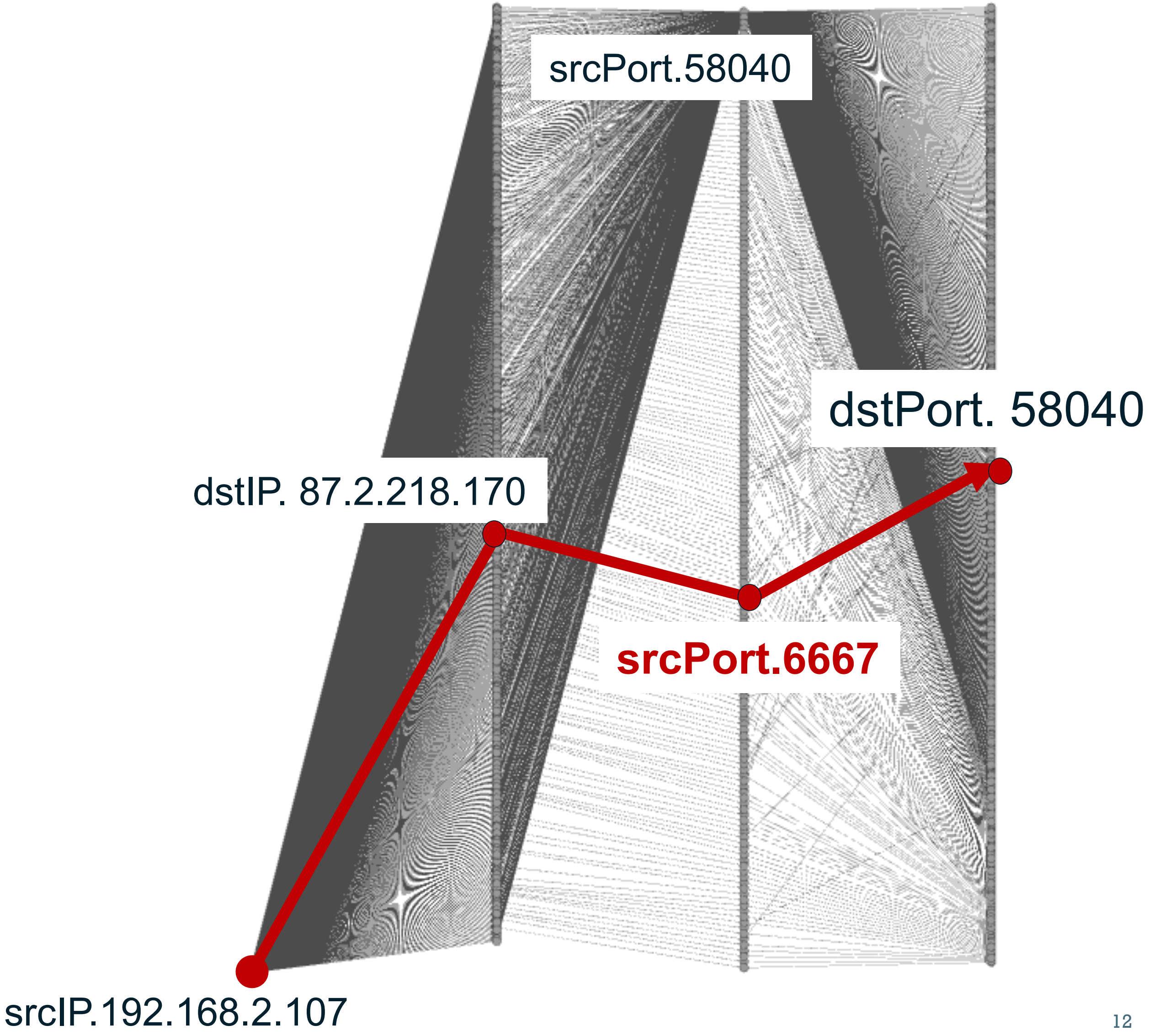


Disambiguation power: Attack vs Normal

Attack SSH



Normal SSH

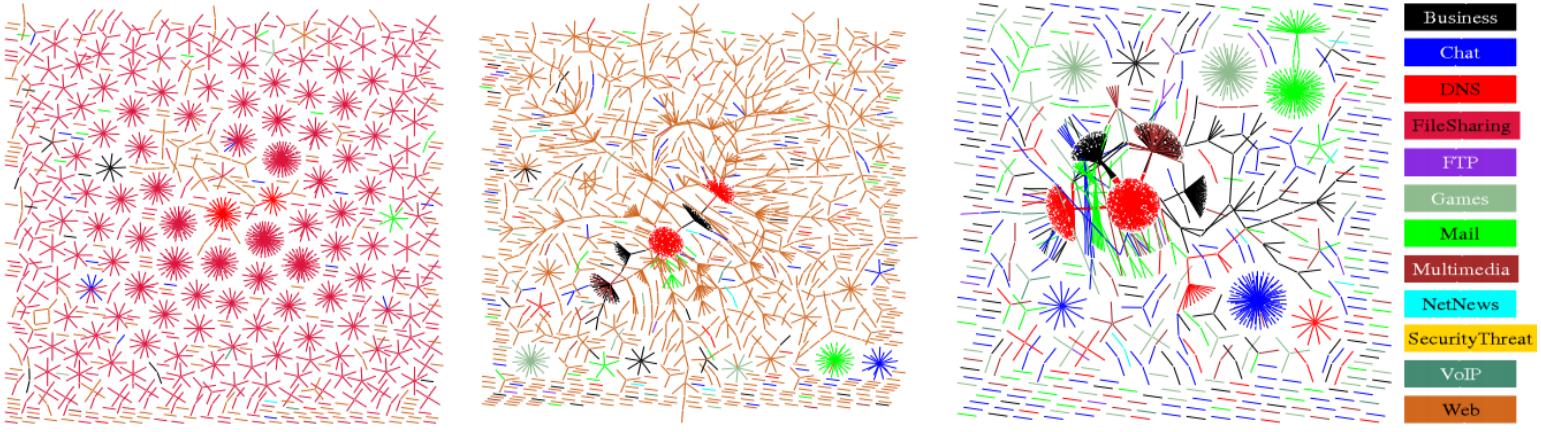


Relational information matters

- Normal and abnormal activities can be detected by chaining packet clustering and analyzing topology of resulting IP-to-IP networks

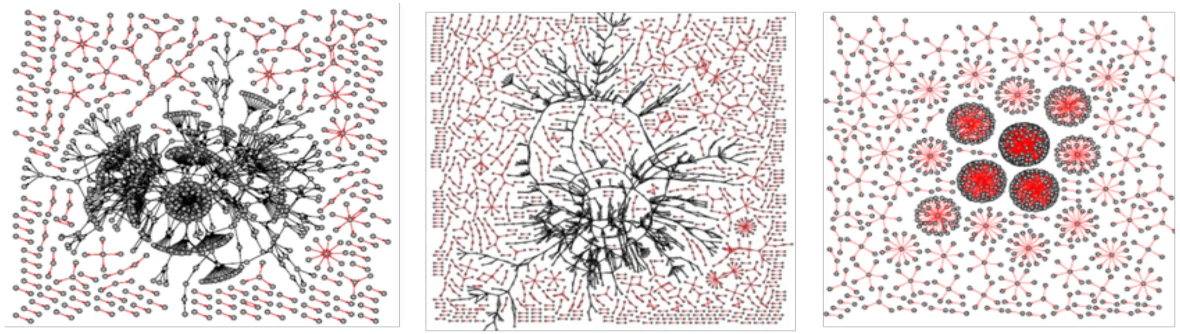
- How much network density do we need to preserve the detection rates?

Function detection from network topology



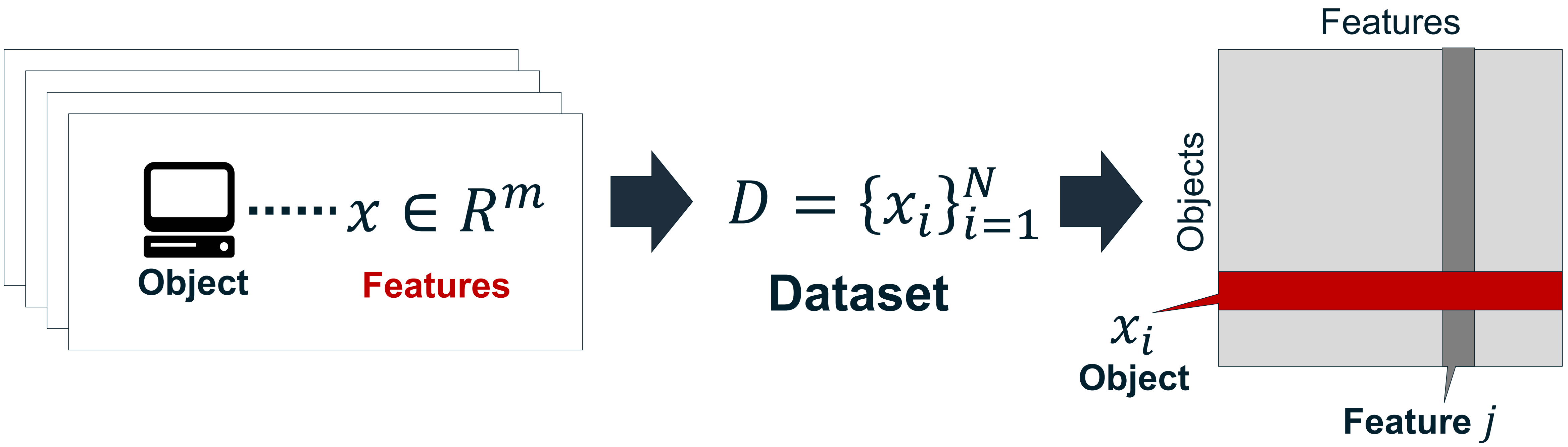
(a) all applications (b) FileSharing removed (c) FileSharing and Web removed

Malware detection from network topology



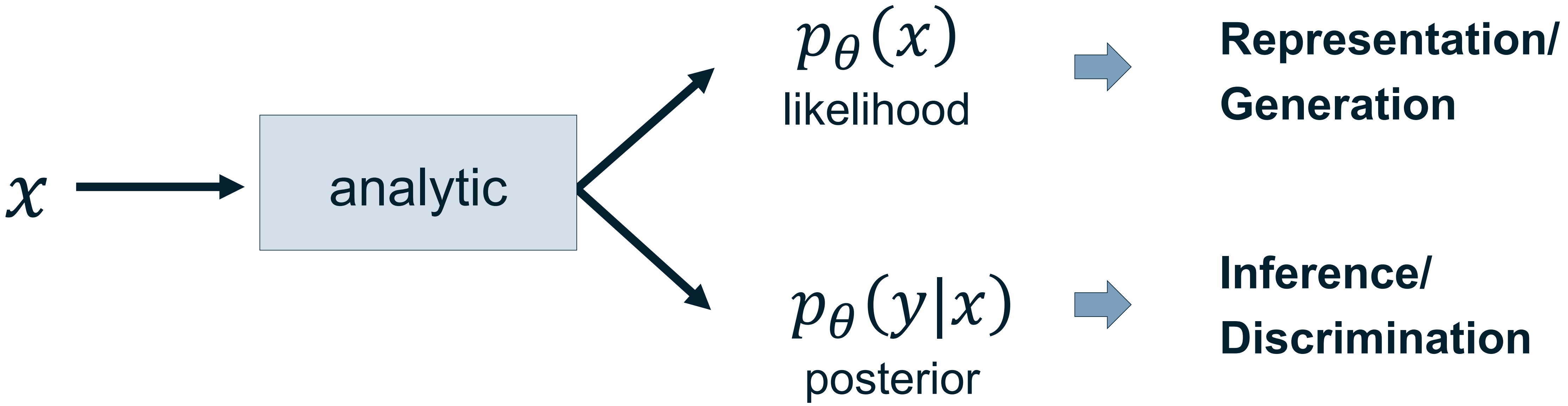
Gnutella P2P	BitTorrent	Slammer worm
High average degree, large connected component	High graph diameter, spider's web with large paths	Large # of discon-nected components, high out-star count

General analysis setup



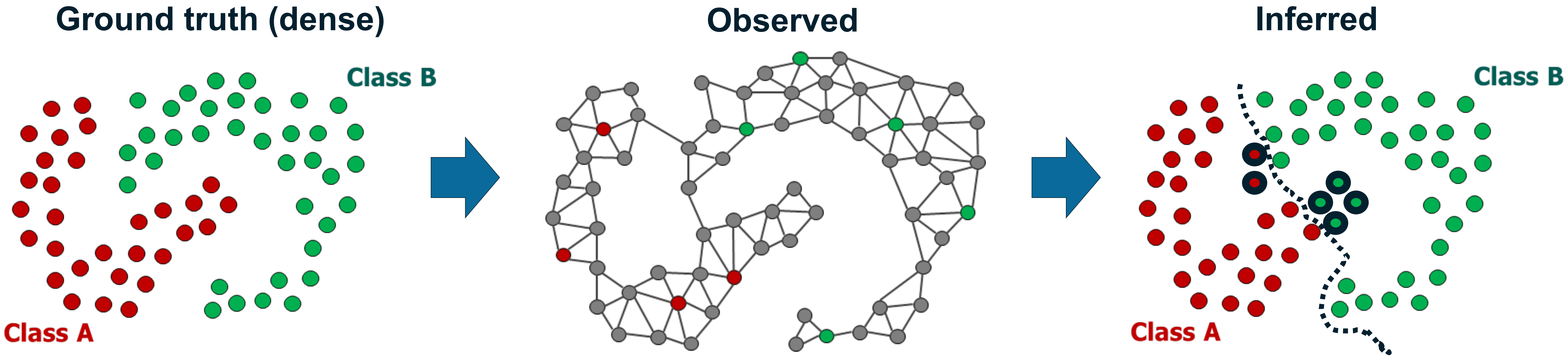
- Dataset can contain very large # of points

General analysis problem



- Technical problems:
  - Learn parameters θ
  - Construct distribution $p_\theta(x)$ or $p_\theta(y|x)$
  - Develop approach to sample from $p_\theta(x)$

Example analytic: semi-supervised learning



- Data contains very few labels
- Graph-based semi-supervised learning exploits structure between unlabeled points
- Label distribution obtained via message passing:

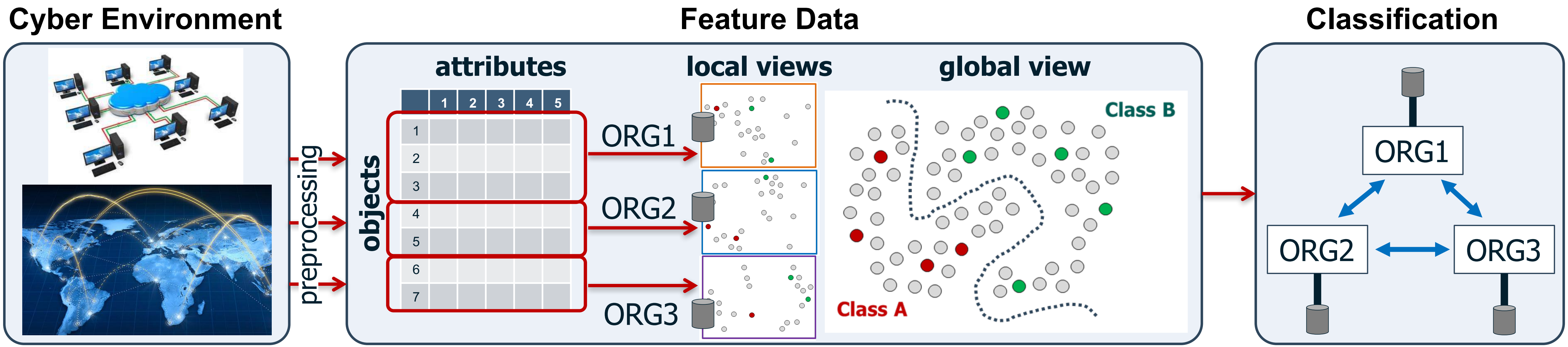
$$y = A \cdot y + z$$

- Closed-form solution:

$$y = (I - A)^{-1} z$$

- Approximate solution via sparse matrix decomposition
  - Has limited scaling

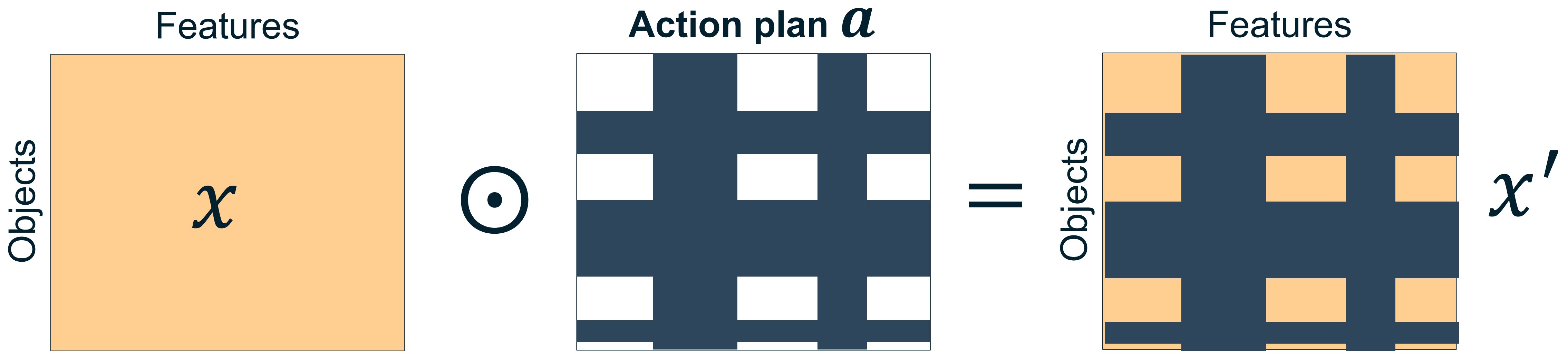
Distributed analysis workflow



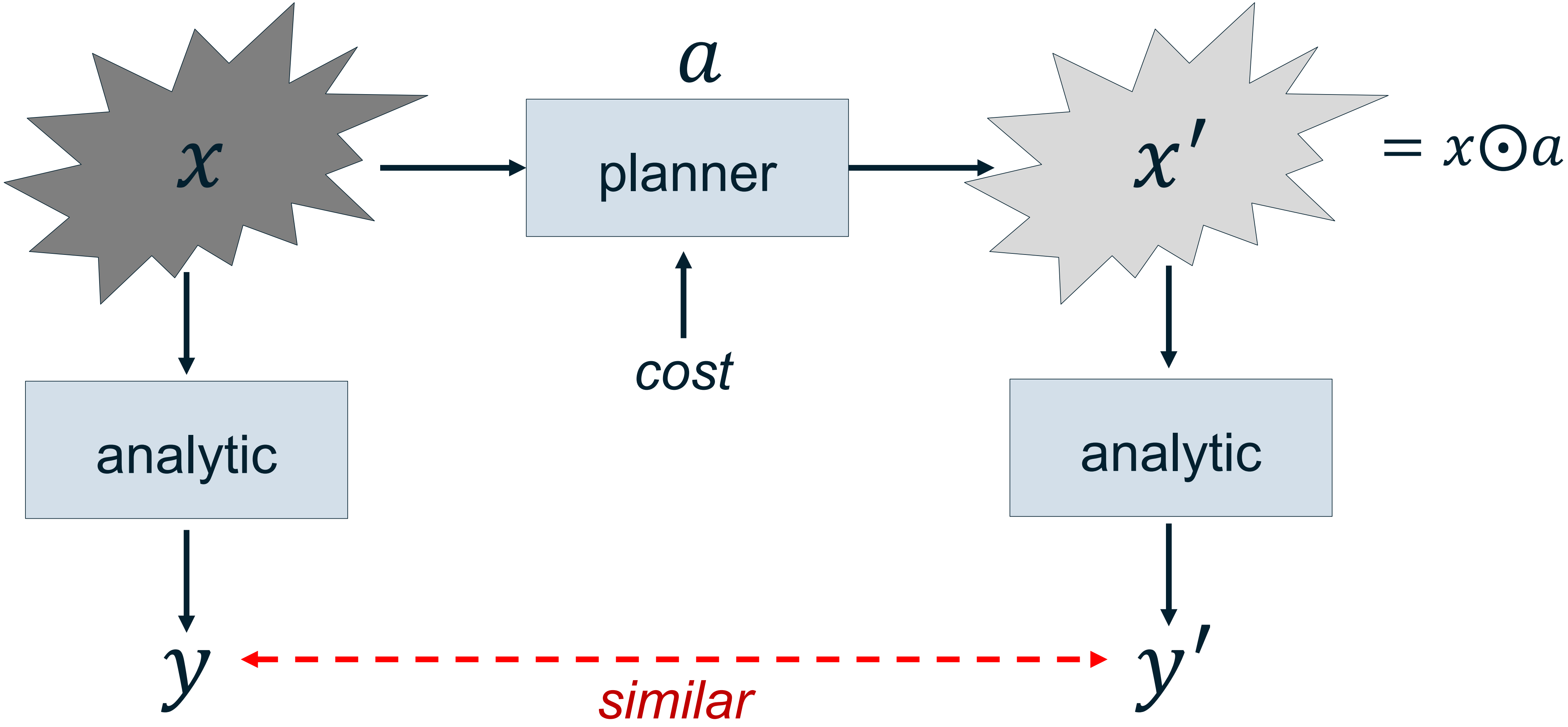
- Distributed processing challenge
  - Local-global data moves restricted
  - Global attacks are locally invisible
  - Analytics chaining/orchestration is ad-hoc
- Data management challenge
  - Multiple analytics have diverse data requirements & goals
  - Individual analytics rarely reason about other analytics

Scale up by filtering

- Generalized representation of objects-features:



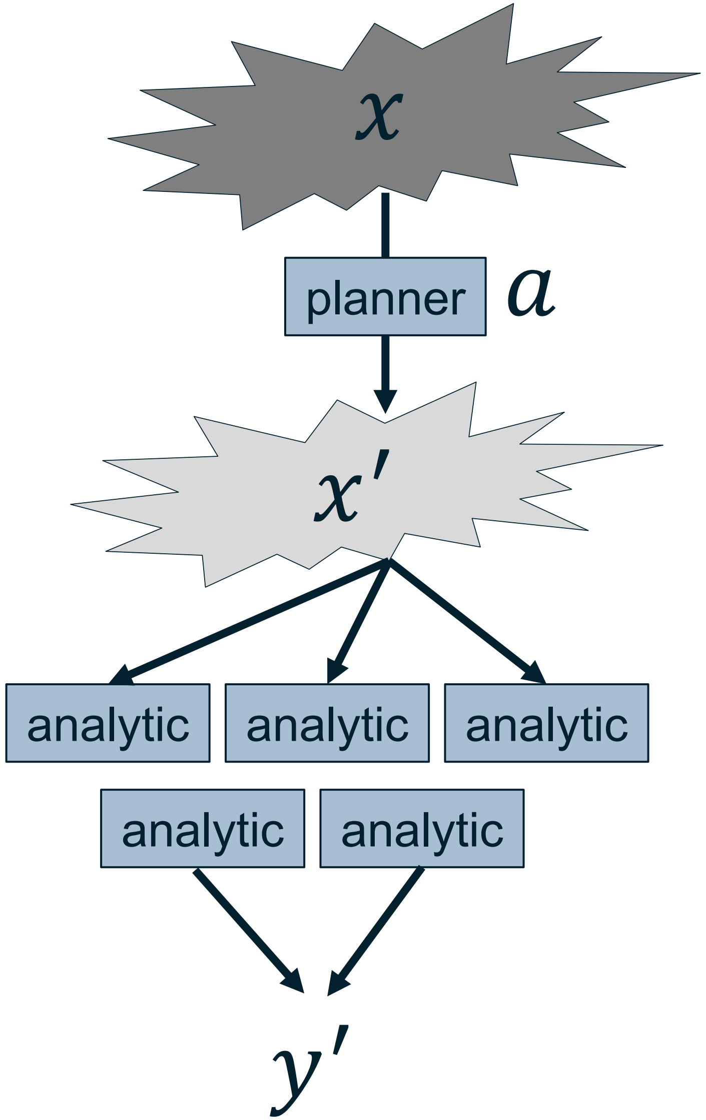
Scale up by filtering



- Planner can define what variables to **collect** or **retain**

Standard solutions

- Feature importance ranking
- Dimensionality reduction
  - PCA
  - Locally linear embedding
  - Manifold learning
- Weaknesses:
  - These solutions are not adaptive to changing environment (variables x) or activities (e.g., attacks)
  - Do not generalize well across domains
  - Cannot be tailored to specific analytics
  - Cannot incorporate costs of data (collection, retention), multiple providers (analytics needing different data), or requests (user needs)



Required workflow

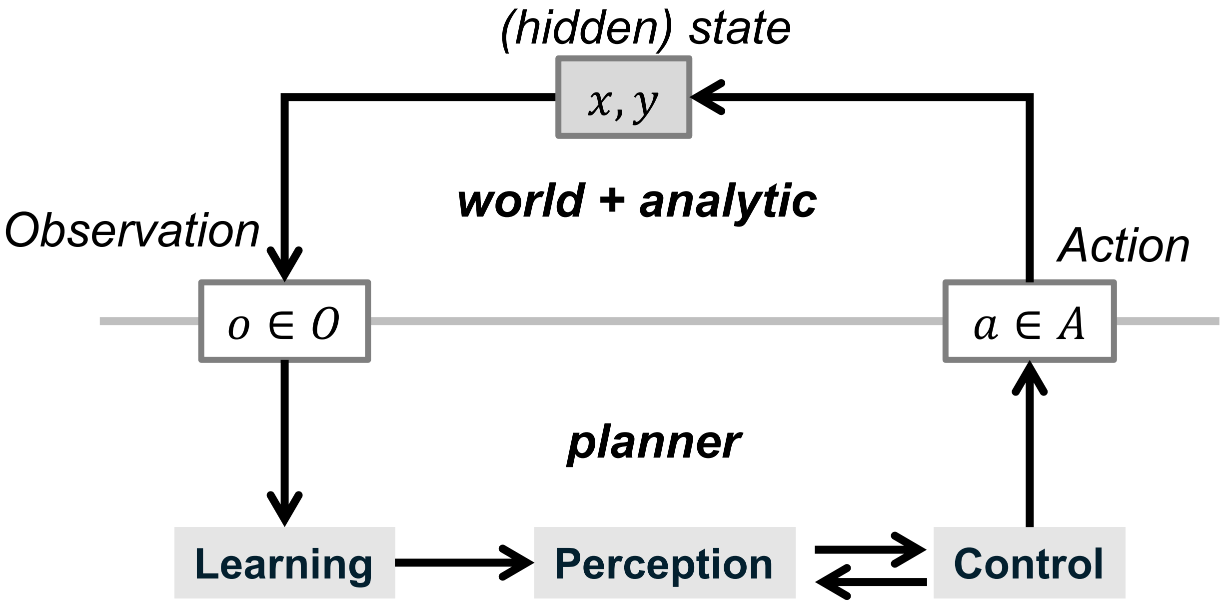
Requirements and solution ideas

- **Requirements:**
  - Can be applied to 1 or more analytics but with unknown “internals”
    - Treat analytics as black-box
  - Can incorporate data costs
  - Can adapt to changing analytic, threat, or environment
  - Can transfer across analytics or domains
  - Can scale to large data sizes

- **Addressed by energy-based variational planning with:**
  - Distribution via restricted Boltzmann machine
    - Simple encoding of pair-wise variable dependencies/ constraints
    - Easy gradient computation
  - Variational bound
    - Avoid costly marginalization
  - Active inference
    - Perception, control, learning cycles
    - Iterate between policy and parameter (reward) learning
    - Policy used to sample actions
  - Scale up via amortized inference & belief propagation

Planning model as "active inference"

- Planner treats analytic(s) as black boxes
- Iteratively samples the space of actions (collection, retention) to learn about the analytic and the world
- Integrates **learning** (parameters), **perception** (about state of the world), and **control** (data action selection)
- Equivalent to inverse reinforcement learnings



Planning model

- Define "outcome success" probability

$$p_\theta(o = 1|x, a) = e^{-c_\theta(x,a)}$$

- Consider **hidden** trajectory dynamics of the "system":

$$\tau = \{(x^t, a^t), t = 1, \dots, T\}$$

- Obtain policy:

$$\pi(a^t|x^t) = \Pr(a^t|x^t, o^{t:T} = 1)$$

- Objective: minimize surprise

$$J(\theta) = \frac{1}{|D|} \sum_{(x)\in D} -\ln p_\theta(x) = E_{(x)\sim D}[c_\theta(x, 1)] + \ln \sum_{(x,a)} e^{-c_\theta(x,a)}$$

- Variational lower bound

$$\mathcal{L}(\theta, q) = E_{(x)\sim D}[c_\theta(x, 1)] - E_{(x,a)\sim q}[c_\theta(x, a)] + H[q]$$

- Problem:

$$\min_\theta \max_q \mathcal{L}(\theta, q) = E_{(x)\sim D}[c_\theta(x, 1)] - E_{(x,a)\sim q}[c_\theta(x, a)] + H[q]$$

The form of "predictive" probability

- The probability distribution must be "simple"
- Use:

$$q(x, a) = q(x)q(a|x)$$

- Then:
  - Learn distribution $q(x)$ from training data D
  - Sample to generate points x
  - Learn distribution $q(a|x)$ using amortized inference
  - Generate samples of points (x, a)
  - Plug into parameter update

Representation

- Recall:

$$p_\theta(o = 1|x, a) = e^{-c_\theta(x,a)}$$

- Cost model:

$$c_\theta(x, a) = b^T x \odot a + (x \odot a)^T W (x \odot a)$$

- Can compute gradient of c_θ:

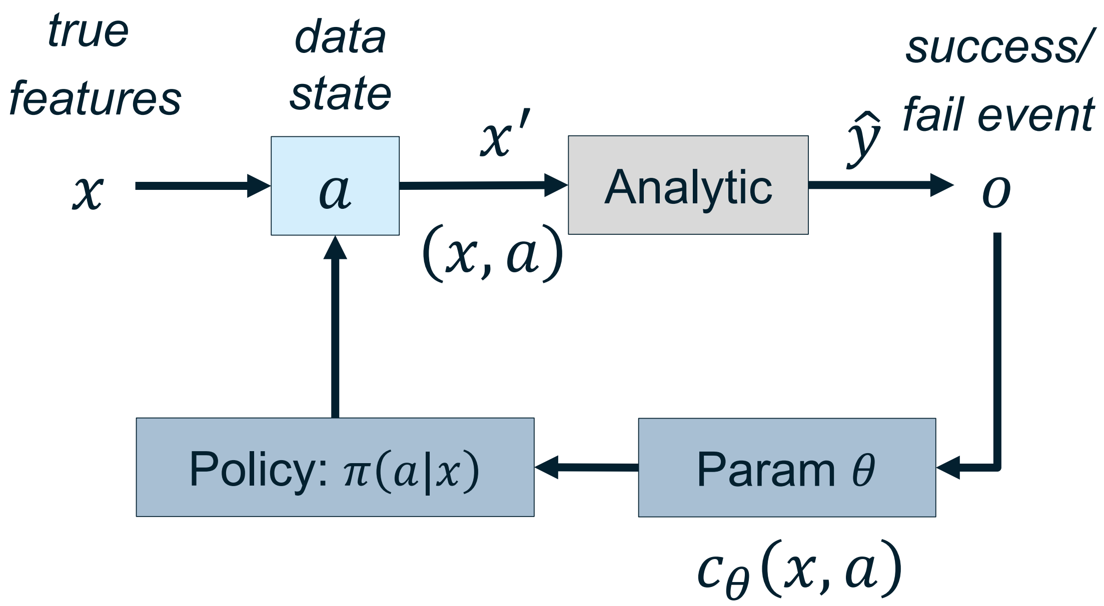
$$\frac{\partial c_\theta(x,a)}{\partial b_i} = x_i a_i, \; \frac{\partial c_\theta(x,a)}{\partial w_{ij}} = x_i a_i x_j a_j$$

- Then parameter updates are simple (error between train data/prior and predictions):

$$b_i \leftarrow b_i - \gamma(x_i - E[x_i a_i])$$
$$w_{ij} \leftarrow w_{ij} - \gamma\left(x_i x_j - E\left[x_i a_i x_j a_j\right]\right)$$

  – In above expectations over marginals (no need for full distribution)
- The control distribution is a form of regularized optimal control, and is solved using soft Q-learning

Planner's recap



Planner:

- Learns parameters θ of cost function:
  $$c_\theta(x, a)$$
- Constructs data plan policy:
  $$\pi(a^t|x^t)$$
- Has intermediate variables as the probability of feature state:
  $$q(x)$$
- Uses parameters of state dynamics:
  $$p(x^{t+1}|x^t, a^t)$$
- Uses the feedback of observed events o
  - Received if can query analytic
  - Difference between predicted and generated values

Why would this be scalable?

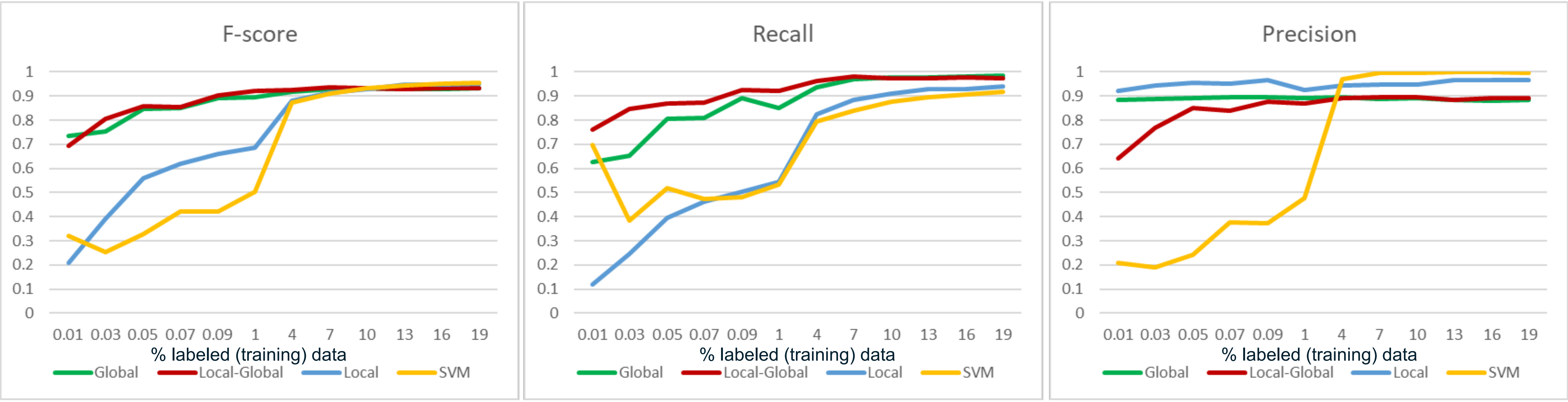
- Can constrain the pair-wise feature correlations to reduce the # of parameters in (and updates of) the matrix W

- Can use alternative methods to estimate generative probability
  - Variational auto-encoders
  - Variational Generative Adversarial networks

- All other updates are linear complexity

Results: sparsity of labeled data

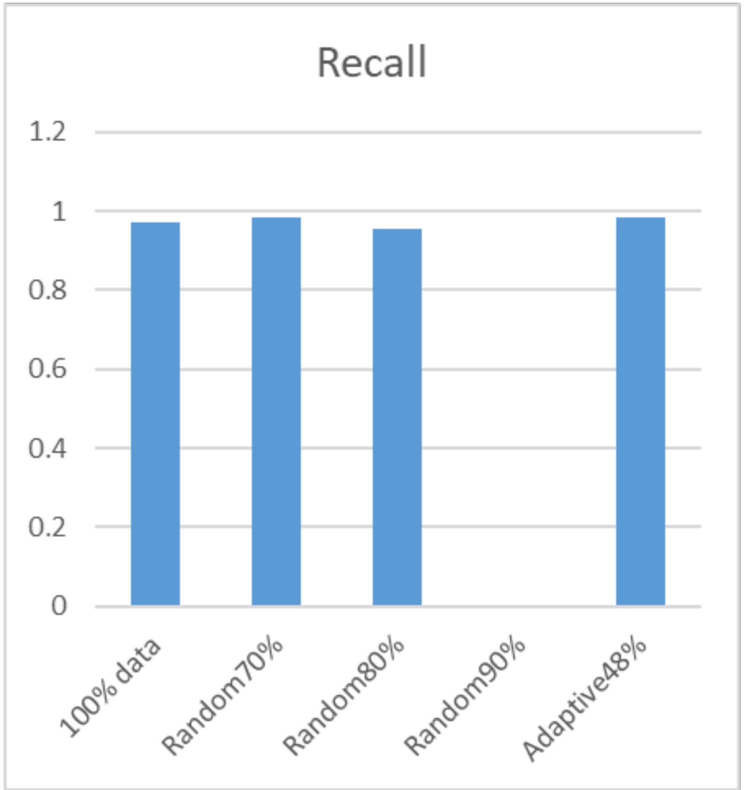
- **Local-Global** (collaborative) semi-supervised algorithm achieved excellent performance (87% Pd, 85% Pf) when only ½% of data points are labeled
  - Matching performance of **global** algorithm
- Neither **local** nor **supervised** classifiers are effective when training (labeled) data is sparse
  - Require 10x (e.g., 10% vs 1%) more labeled examples to match performance of **global** & **local-global** classifiers

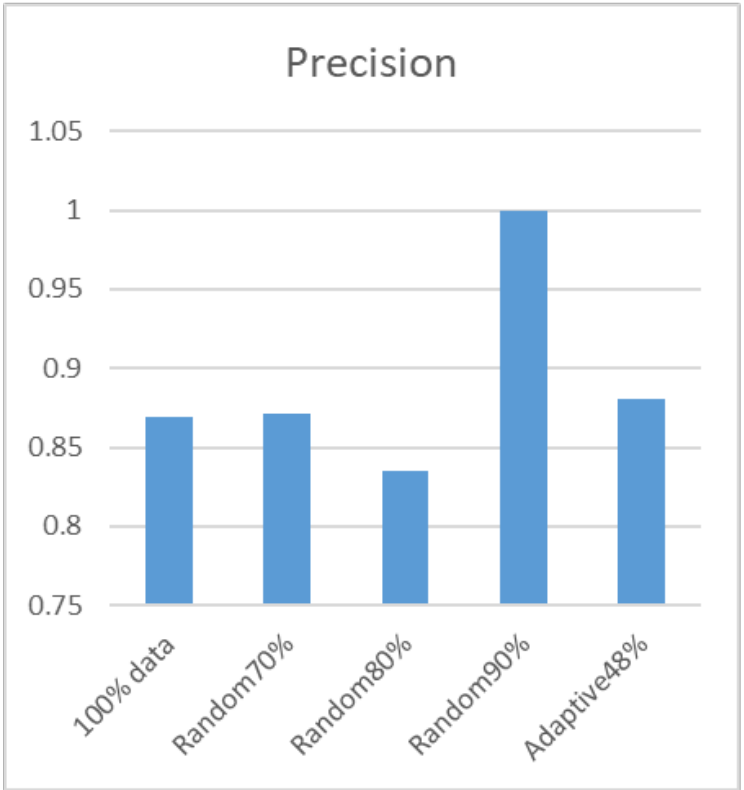


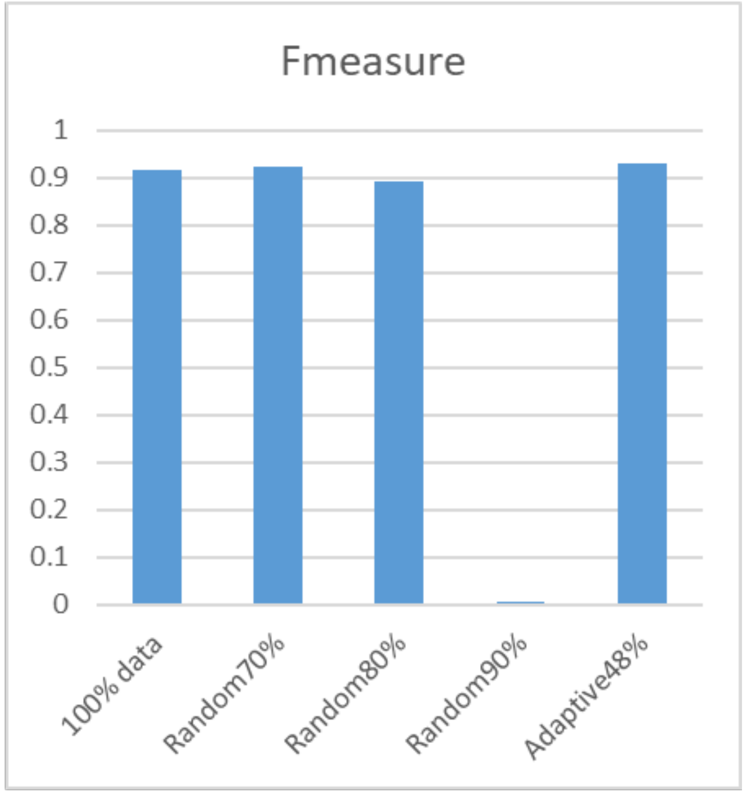
UNSW-NB15 dataset: https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/

Results: sparsity of features

- Adaptive classifier is able to obtain improvement in classification rate by reducing the confusion introduced through redundant and noisy features
- Random feature selection results is drastic reduction of detection quality when significant # of features is removed







Accuracy of classification under different data access conditions

Conclusions

- One of the key methods to improve cyber analytics' performance has always been development of more **meaningful features**
- Introduction of deep machine learning methods promises the discovery of possibly more **discriminative** features, but requires heavy raw data collection
- Current analytics are unable to process the data already being collected, requiring smarter collection planning and retention
- Collection and retention problems can be formalized and solved using similar principles
  - Via adaptive planning
  - Formal approximate solution resembling actor-critic and inverse RL

QUESTIONS?

Georgiy Levchuk |
georgiy@aptima.com
781-496-2467

Aptima, Inc. | www.aptima.com
12 Gill Street, Suite 1400
Woburn, MA 01801