Anomaly Detection: Principles, Benchmarking, Explanation, and Theory

Tom Dietterich
Alan Fern
Weng-Keen Wong

Sharmodeep Battacharyya
Debashis Mondal

Andrew Emmott
Shubhomoy Das
Md. Amran Siddiqui
Zoe Juozapaitis
Si Liu
Khoi Nguyen
Tadesse Zemicheal

Oregon State University

Outline

- Analysis of the Anomaly Detection Problem
- Benchmarking Current Algorithms for Unsupervised AD
- Explaining Anomalies
- Incorporating Expert Feedback
- PAC Theory of Rare Pattern Anomaly Detection

Defining Anomaly Detection

- Data $\{x_i\}_{i=1}^{N}$, each $x_i \in \Re^d$
- Mixture of “nominal” points and “anomaly” points
- Anomaly points are generated by a different generative process than the nominal points

Three Settings

- Supervised
  - Training data labeled with “nominal” or “anomaly”
- Clean
  - Training data are all “nominal”, test data may be contaminated with “anomaly” points.
- Unsupervised
  - Training data consist of mixture of “nominal” and “anomaly” points
  - I will focus on this case

Well-Defined Anomaly Distribution Assumption

- WDAD: the anomalies are drawn from a well-defined probability distribution
  - example: repeated instances of known machine failures

- The WDAD assumption is often risky
  - adversarial situations (fraud, insider threats, cyber security)
  - diverse set of potential causes (novel device failure modes)
  - user's notion of "anomaly" changes with time (e.g., anomaly == "interesting point")

Strategies for Unsupervised Anomaly Detection

- Let α be the fraction of training points that are anomalies
- Case 1: α is large (e.g., > 5%)
  - Fit a 2-component mixture model
    - Requires WDAD assumption
    - Mixture components must be identifiable
    - Mixture components cannot have large overlap in high density regions
- Case 2: α is small (e.g., 1%, 0.1%, 0.01%, 0.001%)
  - Anomaly detection via Outlier detection
    - Does not require WDAD assumption
    - Will fail if anomalies are not outliers (e.g., overlap with nominal density; tightly clustered anomaly density)
    - Will fail if nominal distribution has heavy tails

Outline

- Analysis of the Anomaly Detection Problem
- Benchmarking Current Algorithms for Unsupervised AD
- Explaining Anomalies
- Incorporating Expert Feedback
- PAC Theory of Rare Pattern Anomaly Detection

Benchmarking Study
[Andrew Emmott]

- Most AD papers only evaluate on a few datasets
- Often proprietary or very easy (e.g., KDD 1999)
- Research community needs a large and growing collection of public anomaly benchmarks

Benchmarking Methodology

- Select data sets from UC Irvine repository
  - >= 1000 instances
  - classification or regression
  - <= 200 features
  - numerical features (discrete features ignored)
  - no missing values (mostly)
- Choose one or more classes to be “anomalies”; the rest are “nominals”

Selected Data Sets

Steel Plates Faults
Gas Sensor Array Drift
Image Segmentation
Landsat Satellite
Letter Recognition
OptDigits
Page Blocks
Shuttle
Waveform
Yeast
Abalone
Communities and Crime
Concrete Compressive Strength
Wine
Year Prediction

Systematic Variation of Relevant Aspects

- Point difficulty: How deeply are the anomaly points buried in the nominals?
  - Fit supervised classifier (kernel logistic regression)
  - Point difficulty: $P(\hat{y} = \text{"}nominal\text{"} | x)$ for anomaly points
- Relative frequency:
  - sample from the anomaly points to achieve target values of α
- Clusteredness:
  - greedy algorithm selects points to create clusters or to create widely separated points
- Irrelevant features
  - create new features by random permutation of existing feature values
- Result: 25,685 Benchmark Datasets

Metrics

- AUC (Area Under ROC Curve)
  - ranking loss: probability that a randomly-chosen anomaly point is ranked above a randomly-chosen nominal point
  - transformed value: $\log \frac{AUC}{1-AUC}$
- AP (Average Precision)
  - area under the precision-recall curve
  - average of the precision computed at each ranked anomaly point
  - transformed value: $\log \frac{AP}{\mathbb{E}[AP]} = \log LIFT$

Algorithms

- Density-Based Approaches
  - RKDE: Robust Kernel Density Estimation (Kim & Scott, 2008)
  - EGMM: Ensemble Gaussian Mixture Model (our group)
- Quantile-Based Methods
  - OCSVM: One-class SVM (Schoelkopf, et al., 1999)
  - SVDD: Support Vector Data Description (Tax & Duin, 2004)
- Neighbor-Based Methods
  - LOF: Local Outlier Factor (Breunig, et al., 2000)
  - ABOD: kNN Angle-Based Outlier Detector (Kriegel, et al., 2008)
- Projection-Based Methods
  - IFOR: Isolation Forest (Liu, et al., 2008)
  - LODA: Lightweight Online Detector of Anomalies (Pevny, 2016)

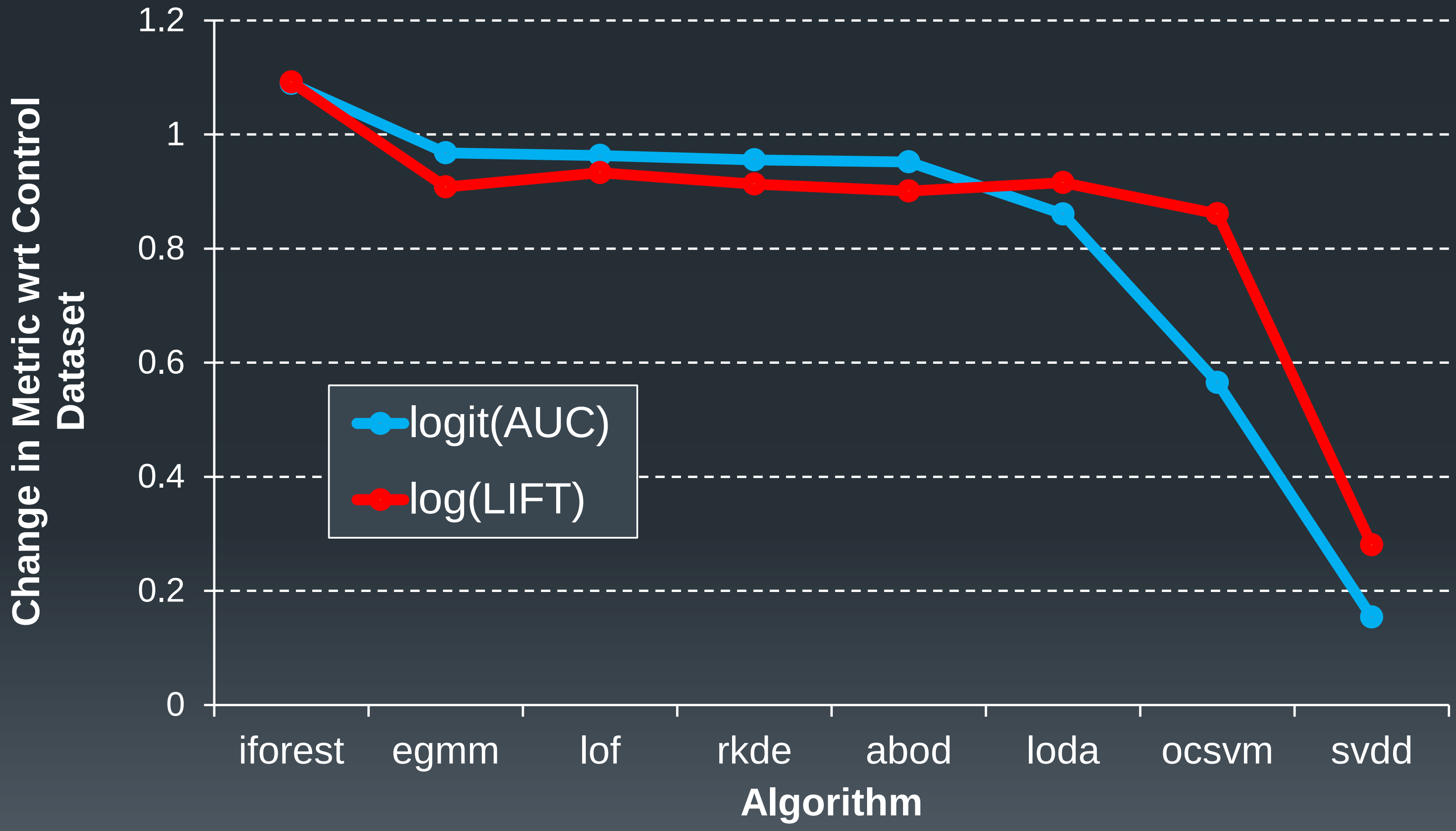
Filtering Out Impossible Benchmarks

- For each algorithm and each benchmark
  - Check whether we can reject the null hypothesis that the achieved AUC (or AP) is better than random guessing
  - If a benchmark dataset is too hard for all algorithms, then we delete it from the benchmark collection

Analysis

- Synthetic Control Data Set
  - Nominals: standard d-dimensional multivariate Gaussian
  - Anomalies: uniform in the $[-4, +4]^d$ hypercube
- Linear ANOVA
  - $metric \sim rf + pd + cl + ir + mset + algo$
    - rf: relative frequency
    - pd: point difficulty
    - cl: normalized clusteredness
    - ir: irrelevant features
    - mset: “Mother” set
    - algo: anomaly detection algorithm
- Assess the *algo* effect while controlling for all other factors

Algorithm Comparison



More Analysis

- In a forthcoming paper, we provide much more detail
  - Mixed-effects model
  - Validation of the importance of each factor
  - Robustness of each algorithm to the factors
- Impact of different factors (descending order)
  - Choice of data set
  - Relative frequency
  - Algorithm
  - Point difficulty
  - Irrelevant features
  - Clusteredness

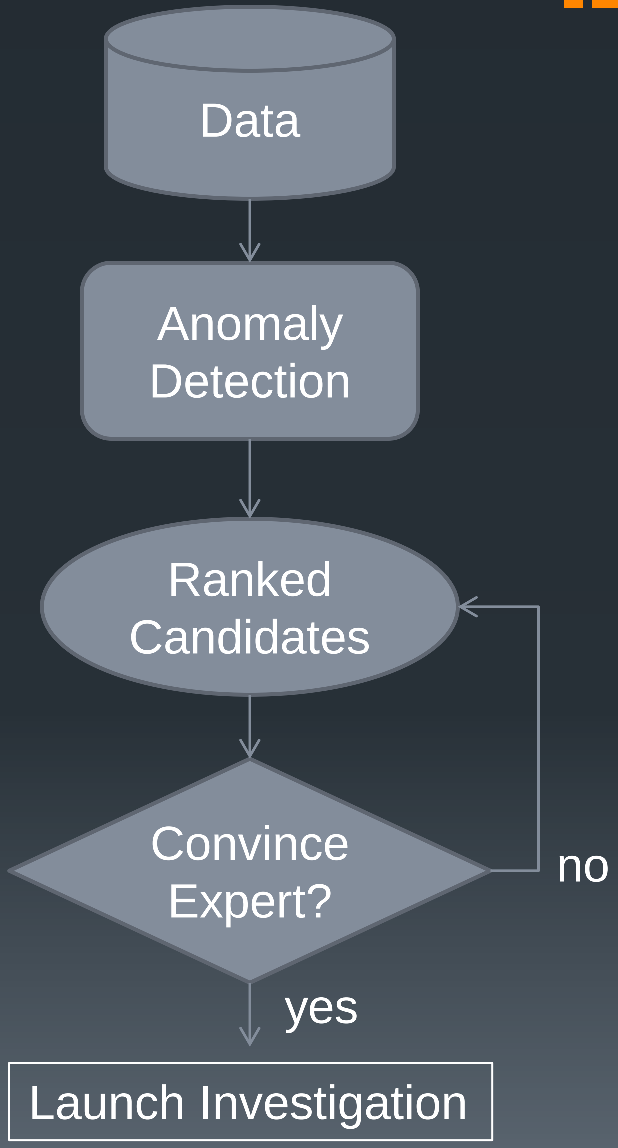
Outline

- Analysis of the Anomaly Detection Problem
- Benchmarking Current Algorithms for Unsupervised AD
- Explaining Anomalies
- Incorporating Expert Feedback
- PAC Theory of Rare Pattern Anomaly Detection

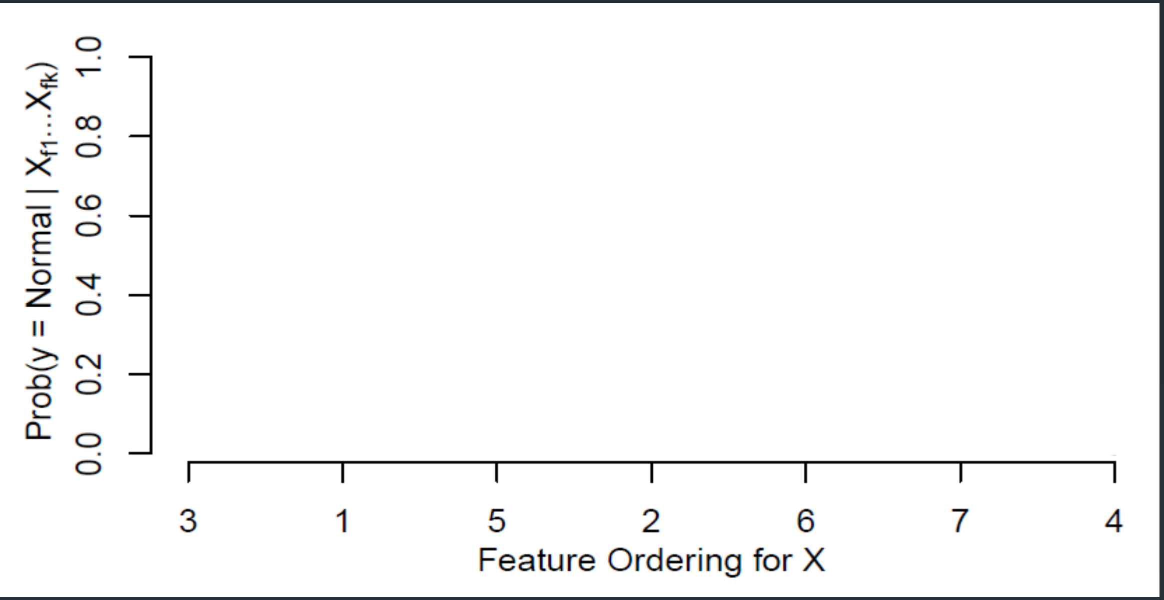
Scenario: Explaining a Candidate Anomaly to an Analyst

- Need to persuade the expert that the candidate anomaly is real
- Idea:
  - Expose one feature value at a time to the expert
  - Provide appropriate visualization tools
- "Sequential Feature Explanation"

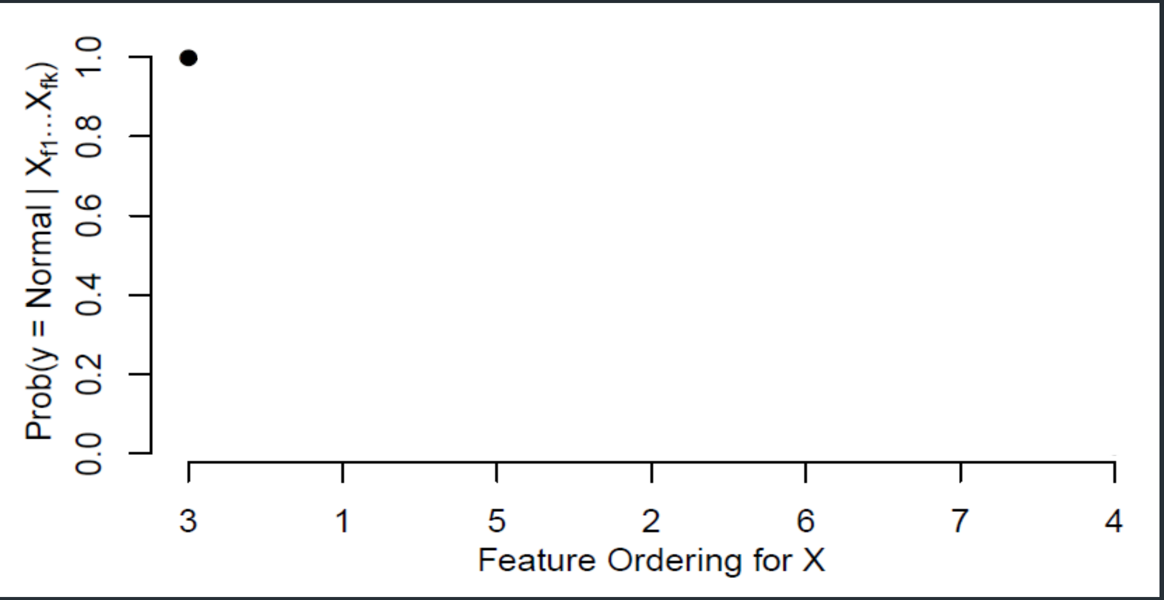
(arXiv:1503.00038)



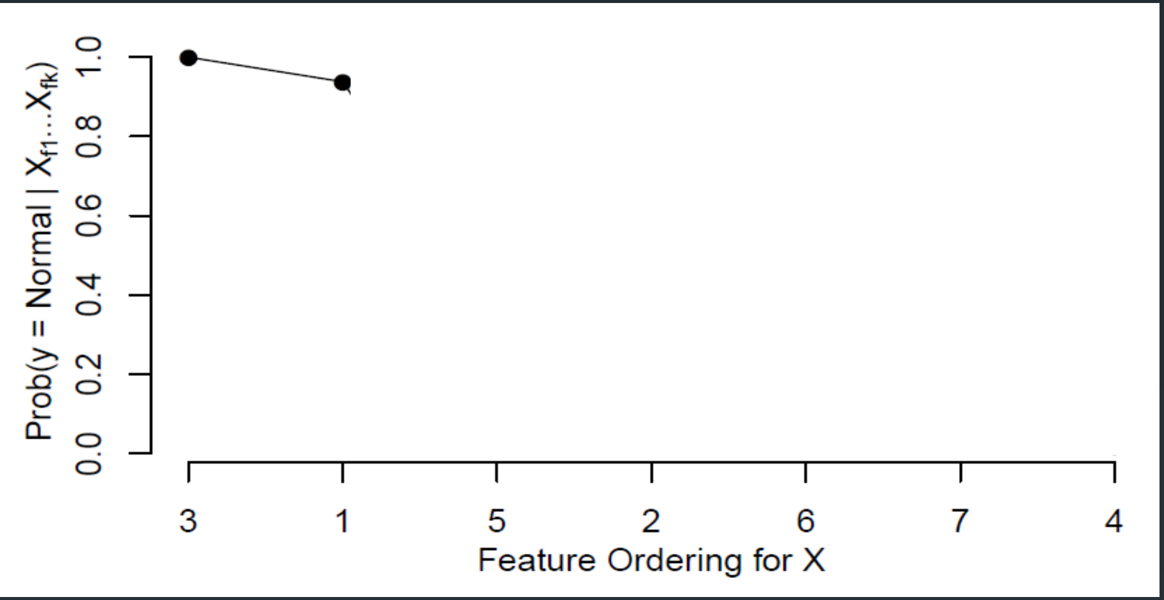
Sequential Feature Explanation (SFE)



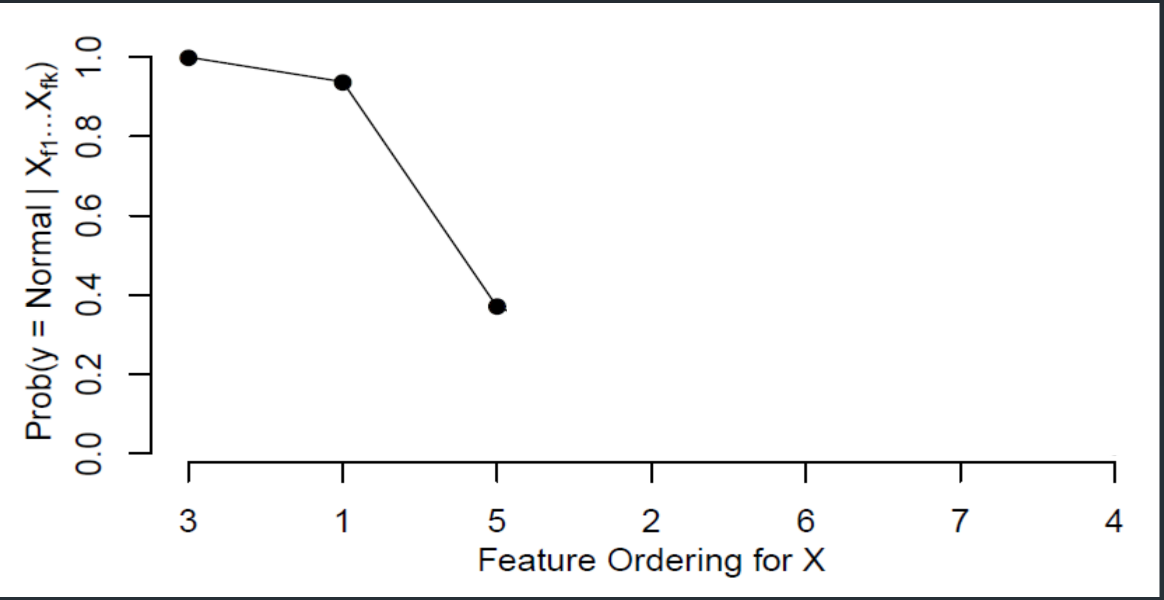
Sequential Feature Explanation (SFE)



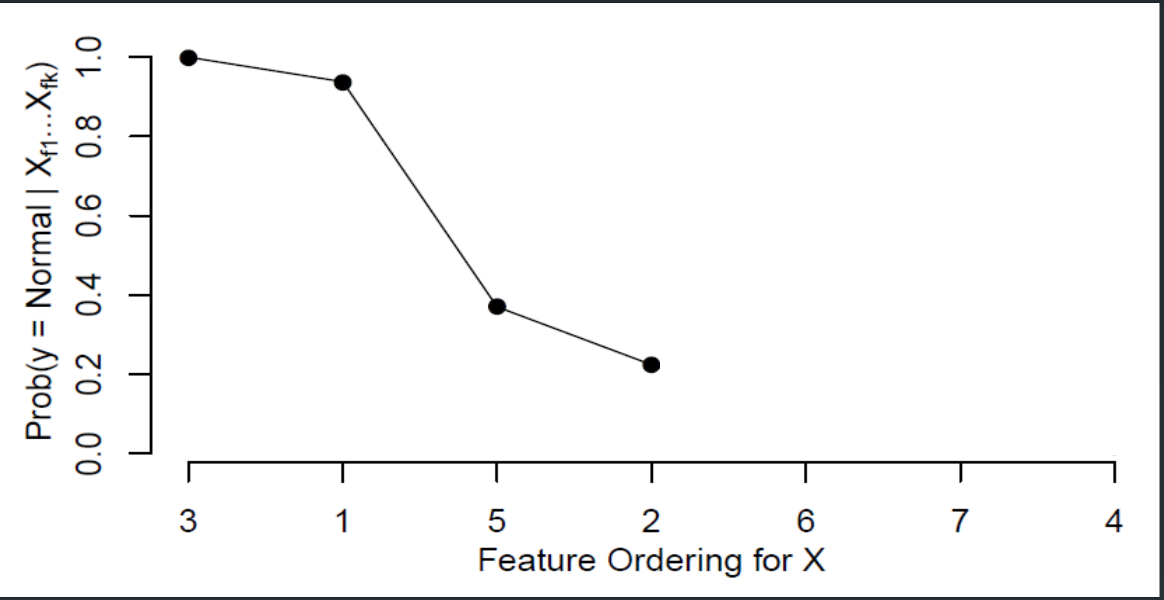
Sequential Feature Explanation (SFE)



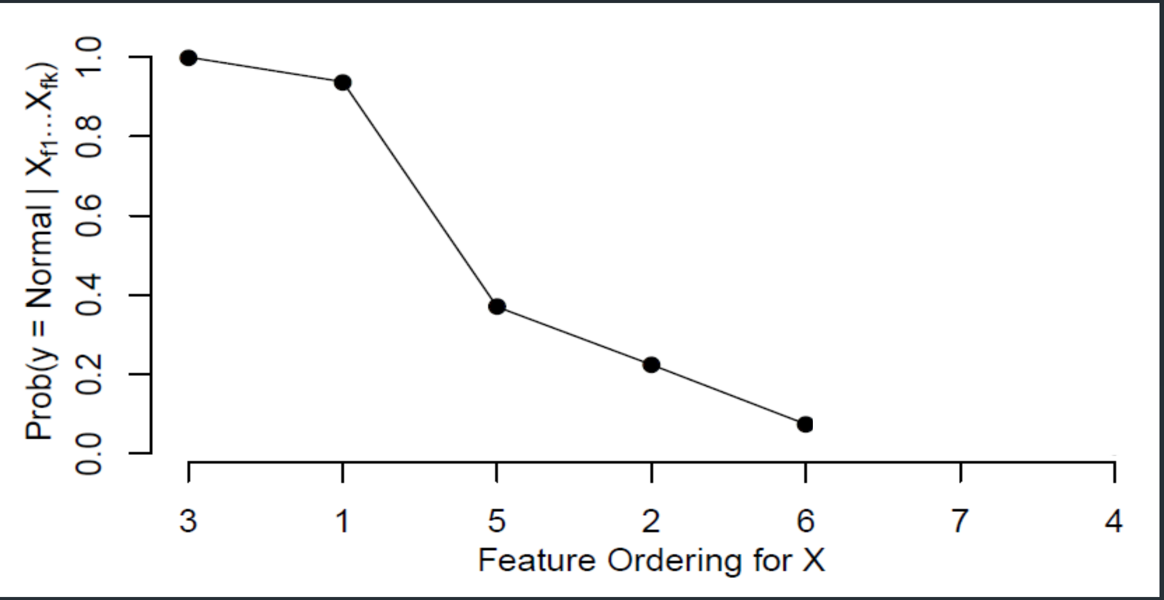
Sequential Feature Explanation (SFE)



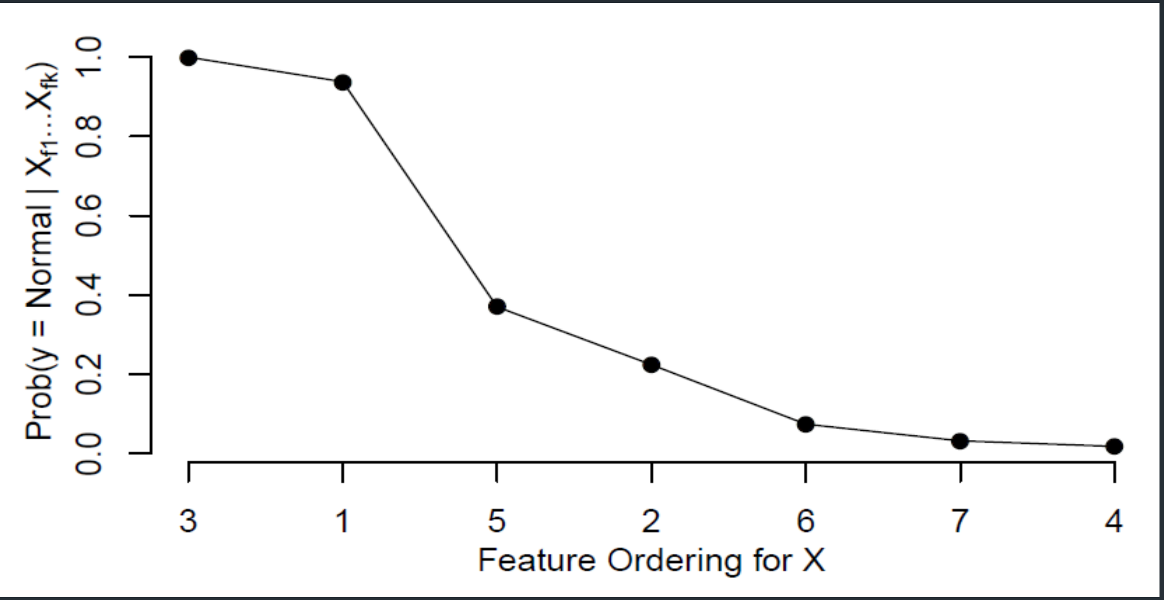
Sequential Feature Explanation (SFE)



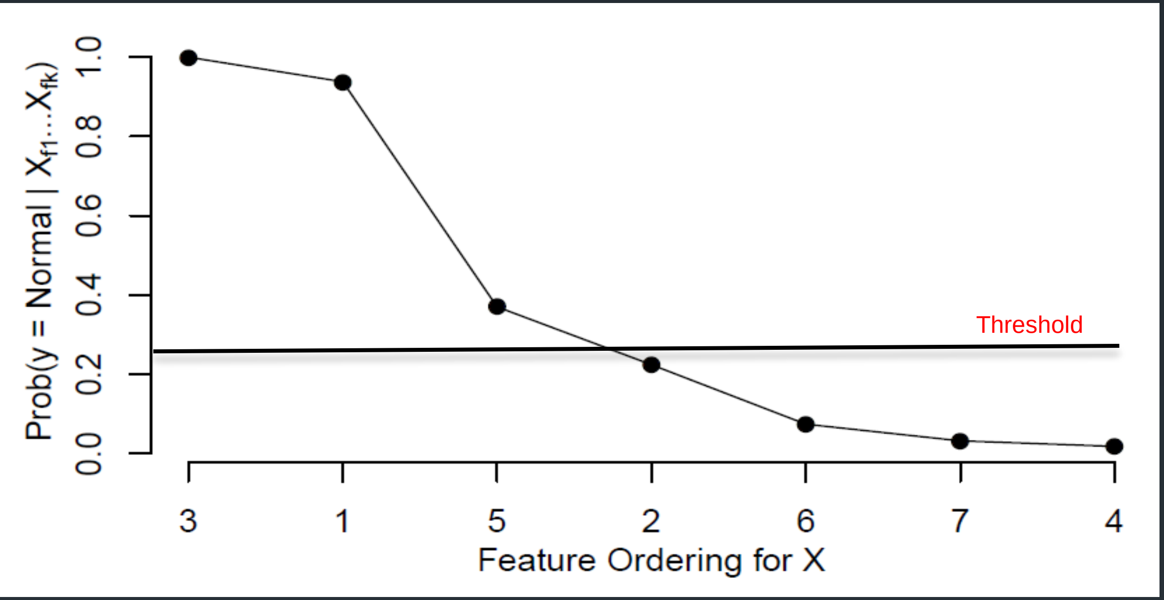
Sequential Feature Explanation (SFE)



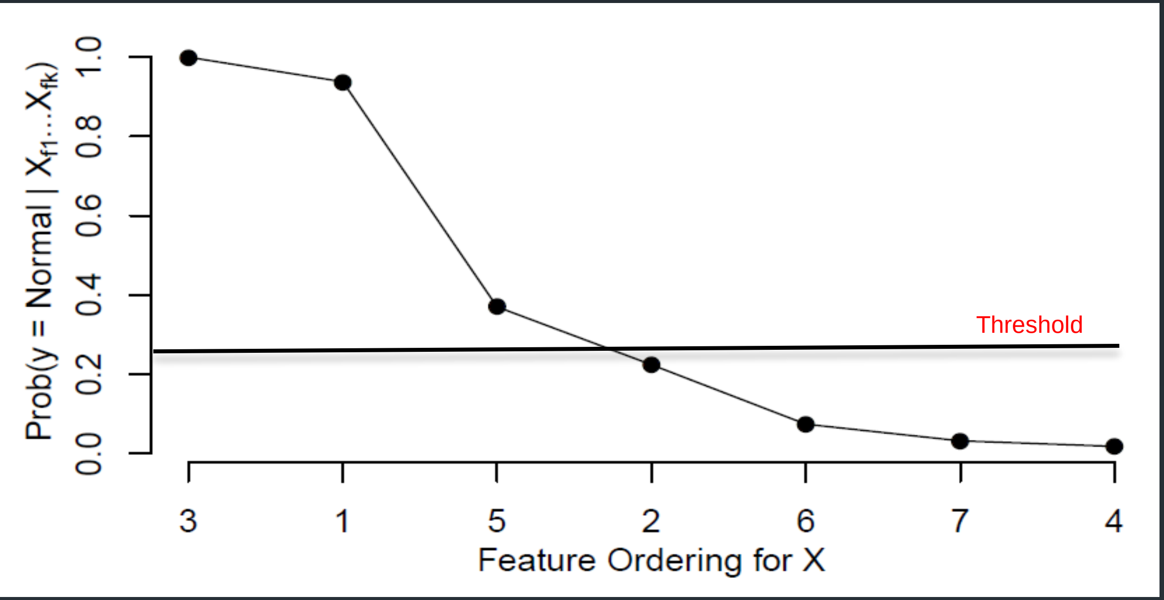
Sequential Feature Explanation (SFE)



Sequential Feature Explanation (SFE)



Sequential Feature Explanation (SFE)



Performance Metric: <u>Minimum Feature Prefix (MFP)</u>. Minimum number of features that must be revealed for the analyst to become confident that a candidate anomaly is a true anomaly. In this example MFP = 4.

Algorithms for Constructing Sequential Feature Explanations [Amran Siddiqui]

- Let $S(x_1, \dots, x_d)$ be the anomaly score for the vector $x = (x_1, \dots, x_d)$
- Assume we have an algorithm that can compute a marginal score for any subset of the dimensions
  - Easy for EGMM, RKDE (score is $-\log \hat{P}(x)$)
- Four Algorithms:

	Marginal	Greedy
Forward Selection	Independent Marginal	Sequential Marginal
Backward Elimination	Independent Dropout	Sequential Dropout

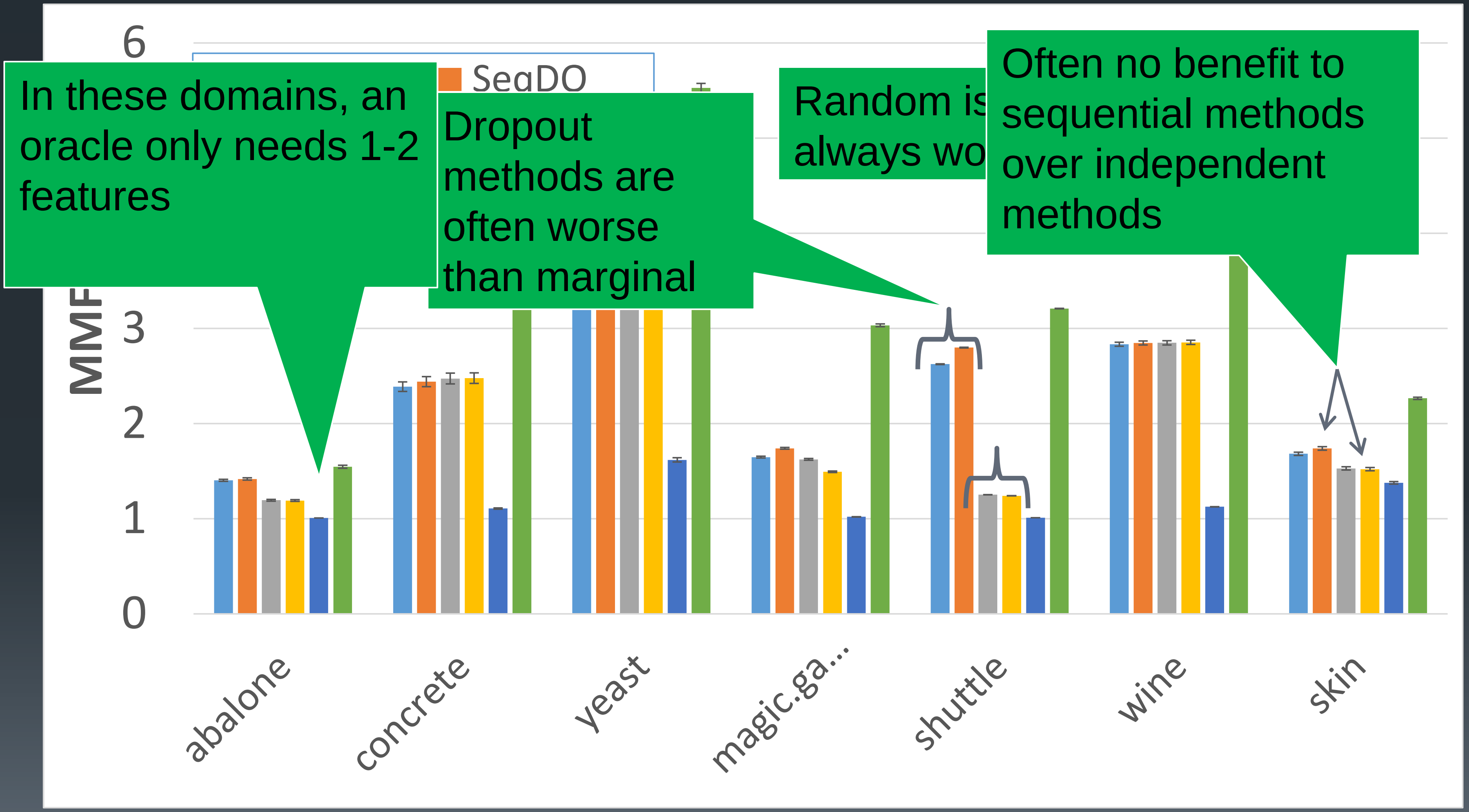
Algorithms

- Independent Marginal
  - Compute $S(x_j)$ for each feature j
  - Order features highest $S\left(x_j\right)$ first
- Sequential Marginal
  - Let $L = \langle\rangle$ be the sequence of features chosen so far
  - Compute $S(L \cup x_j)$ for all $j \notin L$
  - Add the feature j to L that maximizes $S(L \cup x_j)$
- Independent Dropout
  - Let R be the set of all features
  - Compute $S\left(x_{R\setminus\{j\}}\right)$ for each feature j (delete one feature)
  - Sort features lowest $S\left(x_{R\setminus\{j\}}\right)$ first
- Sequential Dropout
  - Let $L = \langle\rangle$ be the sequence of features chosen so far
  - Let R be the set of features not yet chosen
  - Repeat: Add the feature $j \in R$ to L that minimizes $S(x_{R\setminus\{j\}})$

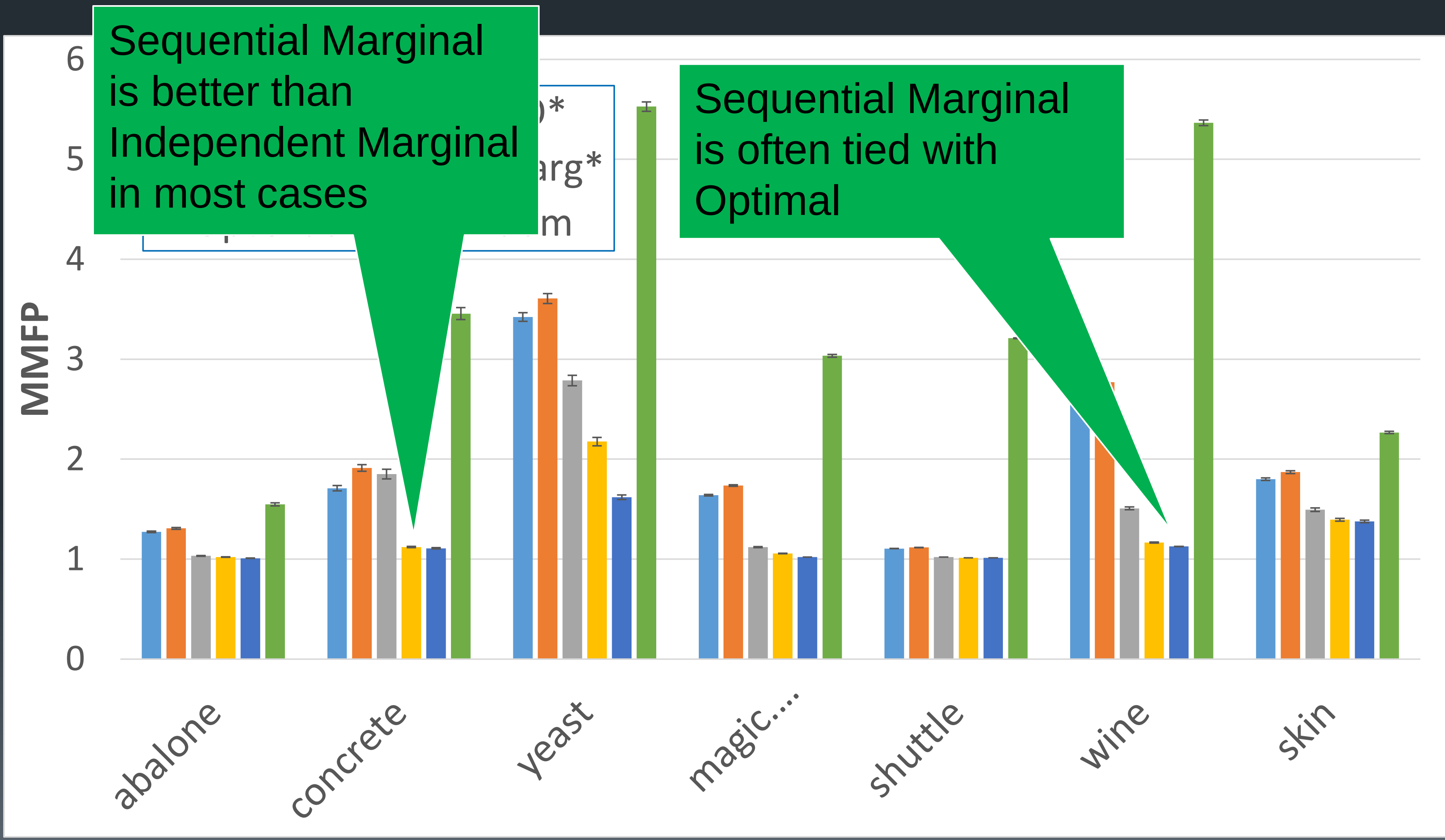
Experimental Evaluations
(1) OSU Anomaly Benchmarks

- **Datasets:** 10,000 benchmarks derived from 7 UCI datasets
- **Anomaly Detector:** Ensemble of Gaussian Mixture Models (EGMM)
- **Simulated Analysts:** Regularized Random Forests (RRFs)
- **Evaluation Metric:** *mean minimum feature prefix (MMFP)* = average number of features revealed on outliers before the analyst is able to make a decision (exonerate vs. open investigation)

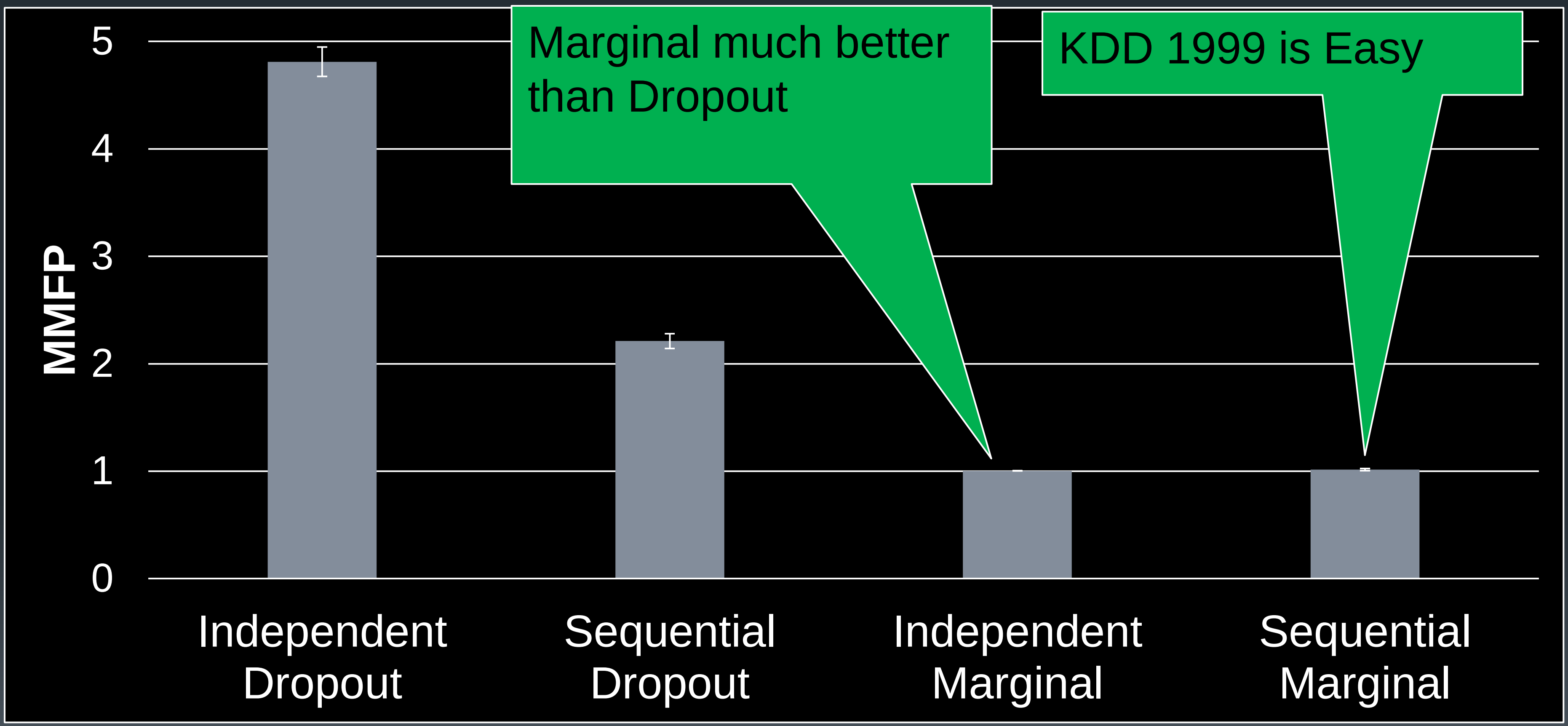
Results (EGMM + Explanation Method)



Results
(Oracle Detector + Explanation Methods)



Experimental Evaluations
(2) KDD 1999 (Computer Intrusion)



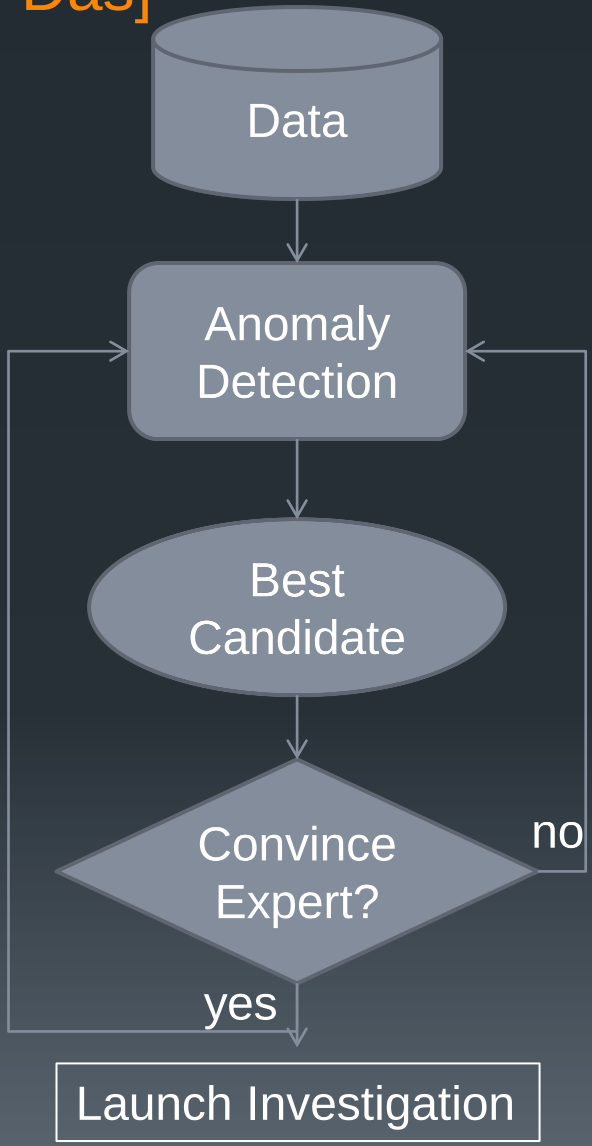
[95% Confidence Intervals]

Outline

- Analysis of the Anomaly Detection Problem
- Benchmarking Current Algorithms for Unsupervised AD
- Explaining Anomalies
- Incorporating Expert Feedback
- PAC Theory of Rare Pattern Anomaly Detection

Incorporating Expert Feedback [Shubhomoy Das]

- Expert labels the best candidate
- Label is used to update the anomaly detector



Idea: Learn to reweight LODA projections

- LODA
  - $\Pi_1, \ldots, \Pi_M$ set of M sparse random projections
  - $f_1, \ldots, f_M$ corresponding 1-dimensional density estimators
  - $S(x) = \frac{1}{M} \sum_m -\log f_m(x)$ average "surprise"
- Parameter τ: quantile corresponding to number of cases analyst can label
- Goal: Learn to reweight the projections so that all known anomalies are above quantile τ and all known nominals are ranked below quantile τ
- Method: Modification of Accuracy-at-the-Top algorithm (Boyd, Mohri, Cortes, Radovanovic, 2012)

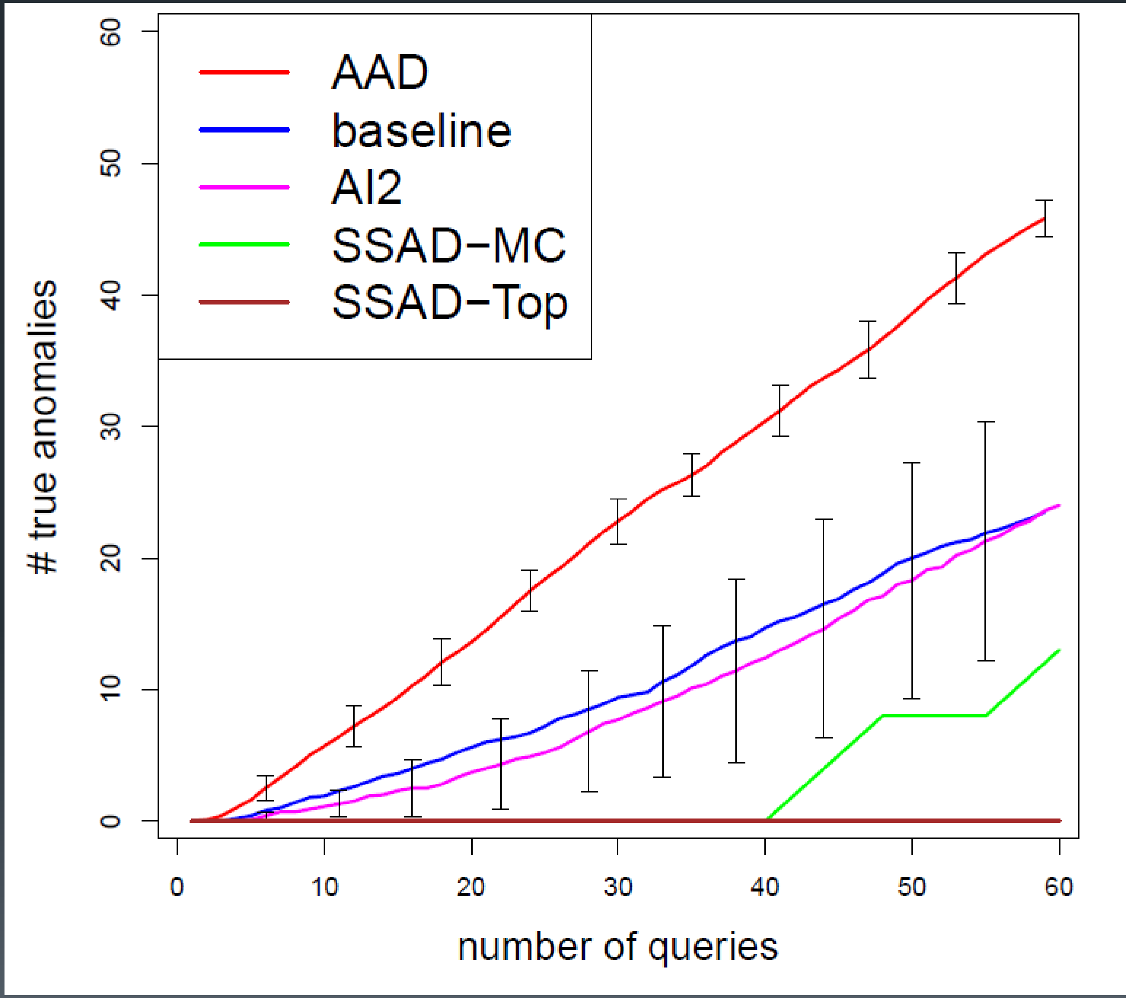
Experimental Setup

Dataset	Nominal Class	Anomaly Class	Total	Dims	# anomalies(%)
Abalone	8, 9, 10	3, 21	1920	9	29 (1.5%)
ANN-Thyroid-1v3	3	1	3251	21	73 (2.25%)
Covtype	2	4	286048	54	2747 (0.9%)
Covtype-sub	2	4	2000	54	19 (0.95%)
KDD-Cup-99	*'normal'*	*'u2r'*, *'probe'*	63009	91	2416 (3.83%)
KDD-Cup-99-sub	*'normal'*	*'u2r'*, *'probe'*	2000	91	77 (3.85%)
Mammography	-1	+1	11183	6	260 (2.32%)
Mammography-sub	-1	+1	2000	6	46 (2.3%)
Shuttle	1	2, 3, 5, 6, 7	12345	9	867 (7.02%)
Shuttle-sub	1	2, 3, 5, 6, 7	2000	9	140 (7.0%)
Yeast	*'CYT'*, *'NUC'*, *'MIT'*	*'ERL'*, *'POX'*, *'VAC'*	1191	8	55 (4.6%)

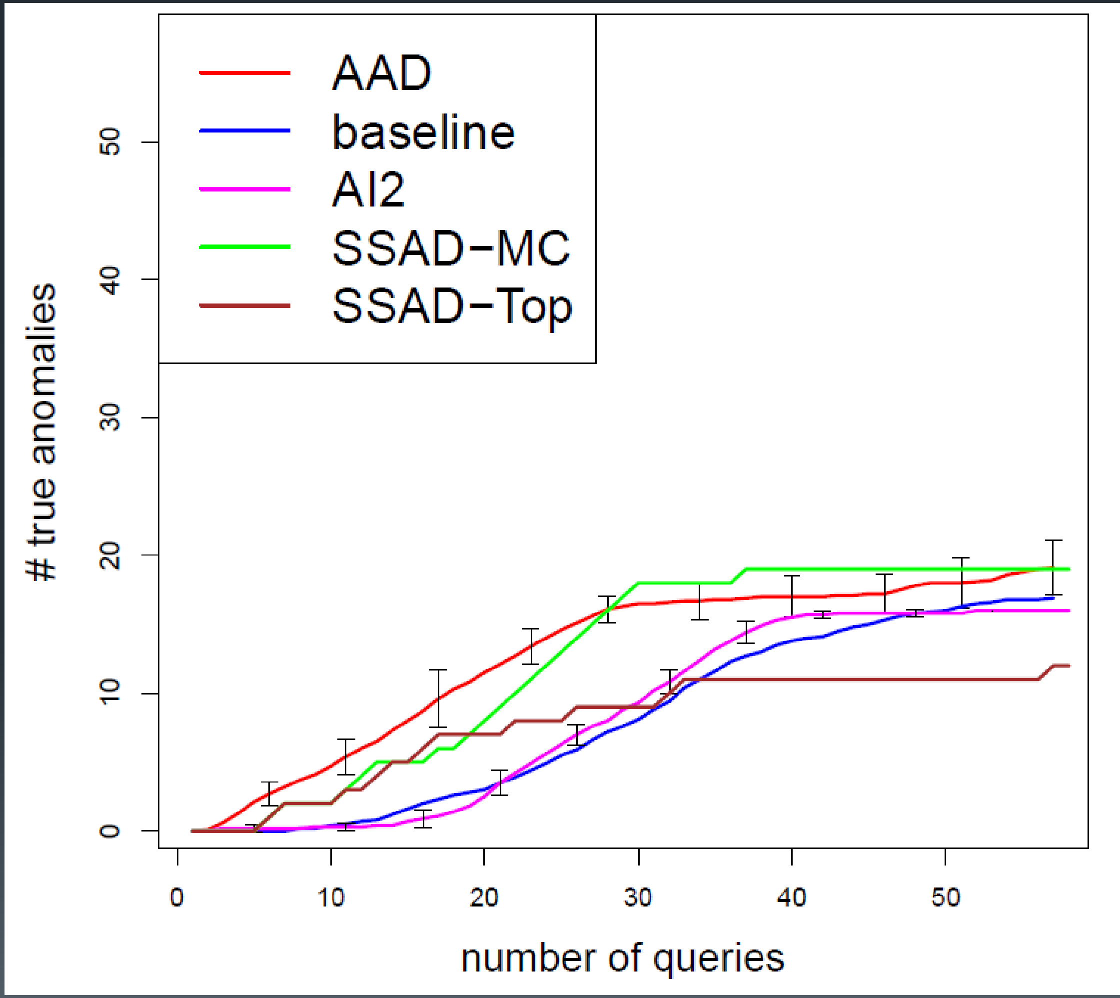
Algorithms

- Baseline: No learning; order cases highest $S(x)$ first
- Random: order cases at random
- AAD: Our method
- AI2: Veeramachaneni, et al. (CSAIL TR).
- SSAD: Semi-Supervised Anomaly Detector (Görnitz, et al., JAIR 2013)

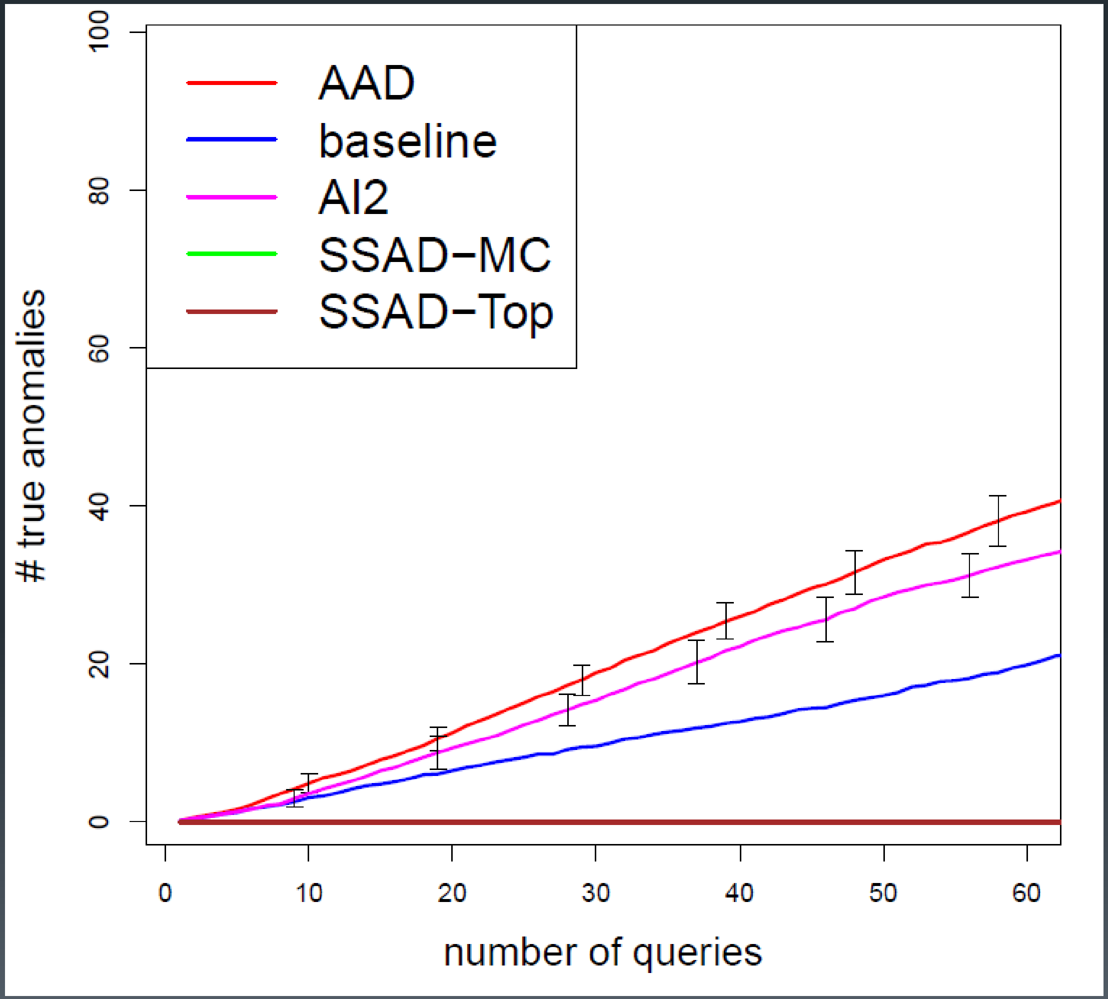
Results: KDD 1999



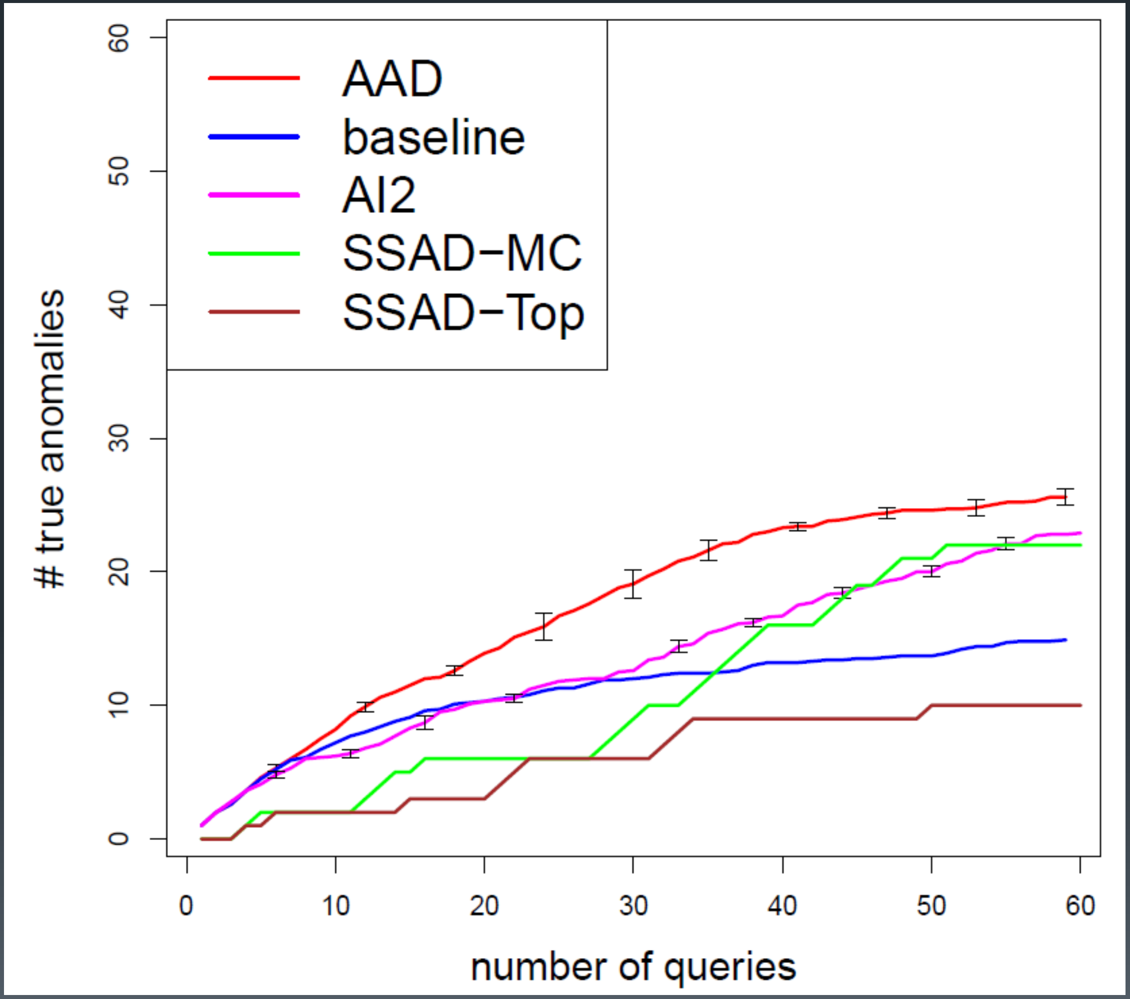
Results: Abalone



Results: ANN-Thyroid-1v3



Results: Mammography



Summary: Incorporating Expert Feedback

- This can be very successful with LODA
  - Even when the expert labels the initial candidates as “nominal”
- AAD is doing implicit feature selection

Outline

- Analysis of the Anomaly Detection Problem
- Benchmarking Current Algorithms for Unsupervised AD
- Explaining Anomalies
- Incorporating Expert Feedback
- PAC Theory of Rare Pattern Anomaly Detection

Towards a Theory of Anomaly Detection [Siddiqui, et al.; UAI 2016]

- Existing theory on sample complexity
  - Density Estimation Methods:
    - Exponential in the dimension d
  - Quantile Methods (OCSVM and SVDD):
    - Polynomial sample complexity
- Experimentally, many anomaly detection algorithms learn very quickly (e.g., 500-2000 examples)
- New theory: Rare Pattern Anomaly Detection

Pattern Spaces

- A pattern $h: \Re^d \rightarrow \{0,1\}$ is an indicator function for a measurable region in the input space
  - Examples:
    - Half planes
    - Axis-parallel hyper-rectangles in $[-1,1]^d$
- A pattern space $\mathcal{H}$ is a set of patterns (countable or uncountable)

Rare and Common Patterns

- Let μ be a fixed measure over $\Re^d$
  - Typical choices:
    - uniform over $[-1, +1]^d$
    - standard Gaussian over $\Re^d$
- $\mu(h)$ is the measure of the pattern defined by h
- Let p be the “nominal” probability density defined on $\Re^d$ (or on some subset)
- $p(h)$ is the probability of pattern h
- A pattern h is τ-rare if

$$f(h) = \frac{p(h)}{\mu(h)} \leq \tau$$

- Otherwise it is τ-common

Rare and Common Points

- A point x is τ-rare if there exists a τ-rare h such that $h(x) = 1$
- Otherwise a point is τ-common

- Goal: An anomaly detection algorithm should output all τ-rare points and not output any τ-common points

PAC-RPAD

- Algorithm $\mathcal{A}$ is PAC-RPAD with parameters τ, ϵ, δ if for any probability density p and any τ, with probability $1 - \delta$ over samples drawn from p, $\mathcal{A}$ draws a sample from p and detects all τ-outliers and rejects all $(\tau + \epsilon)$-commons in the sample

- ϵ allows the algorithm some margin for error
- If a point is between τ-rare and $(\tau + \epsilon)$-common, the algorithm can treat it arbitrarily

RarePatternDetect

- Draw a sample of size $N(\epsilon, \delta)$ from p
- Let $\hat{p}(h)$ be the fraction of sample points that satisfy h
- Let $\hat{f}(h) = \frac{\hat{p}(h)}{\mu(h)}$ be the estimated rareness of h
- A query point x_q is declared to be an anomaly if there exists a pattern $h \in \mathcal{H}$ such that $h(x_q) = 1$ and $\hat{f}(h) \leq \tau$.

Results

- Theorem 1: For any finite pattern space $\mathcal{H}$, RarePatternDetect is PAC-RPAD with sample complexity

$$N(\epsilon, \delta) = O\left(\frac{1}{\epsilon^2}\left(\log|\mathcal{H}| + \log\frac{1}{\delta}\right)\right)$$

- Theorem 2: For any pattern space $\mathcal{H}$ with finite VC dimension $\mathcal{V}_{\mathcal{H}}$, RarePatternDetect is PAC-RPAD with sample complexity

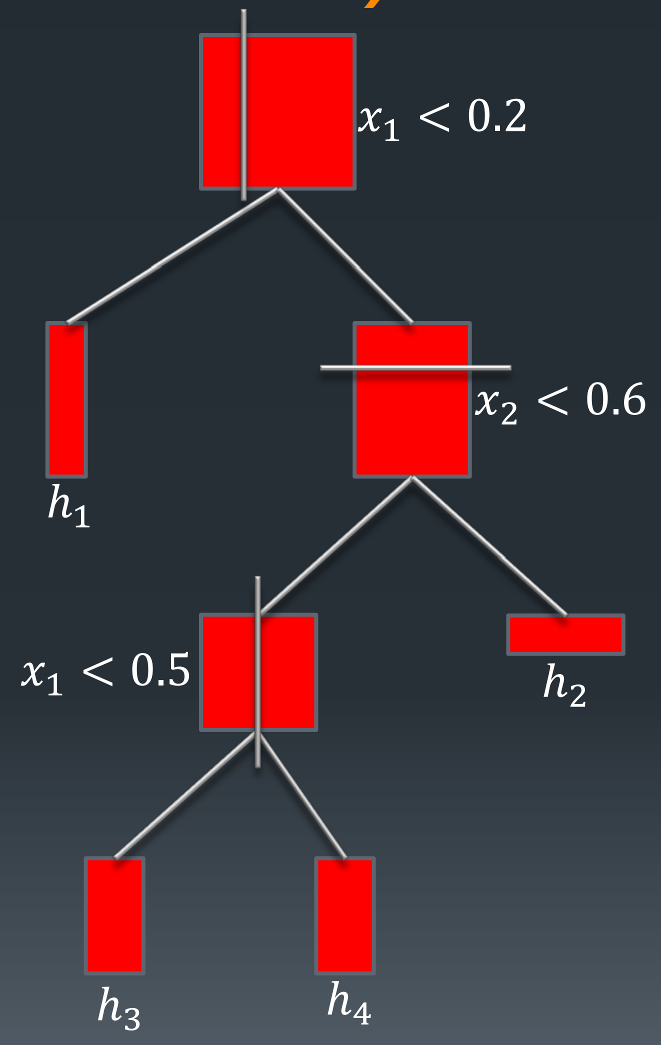
$$N(\epsilon, \delta) = O\left(\frac{1}{\epsilon^2}\left(\mathcal{V}_{\mathcal{H}} \log\frac{1}{\epsilon^2} + \log\frac{1}{\delta}\right)\right)$$

Examples of PAC-RPAD $\mathcal{H}$

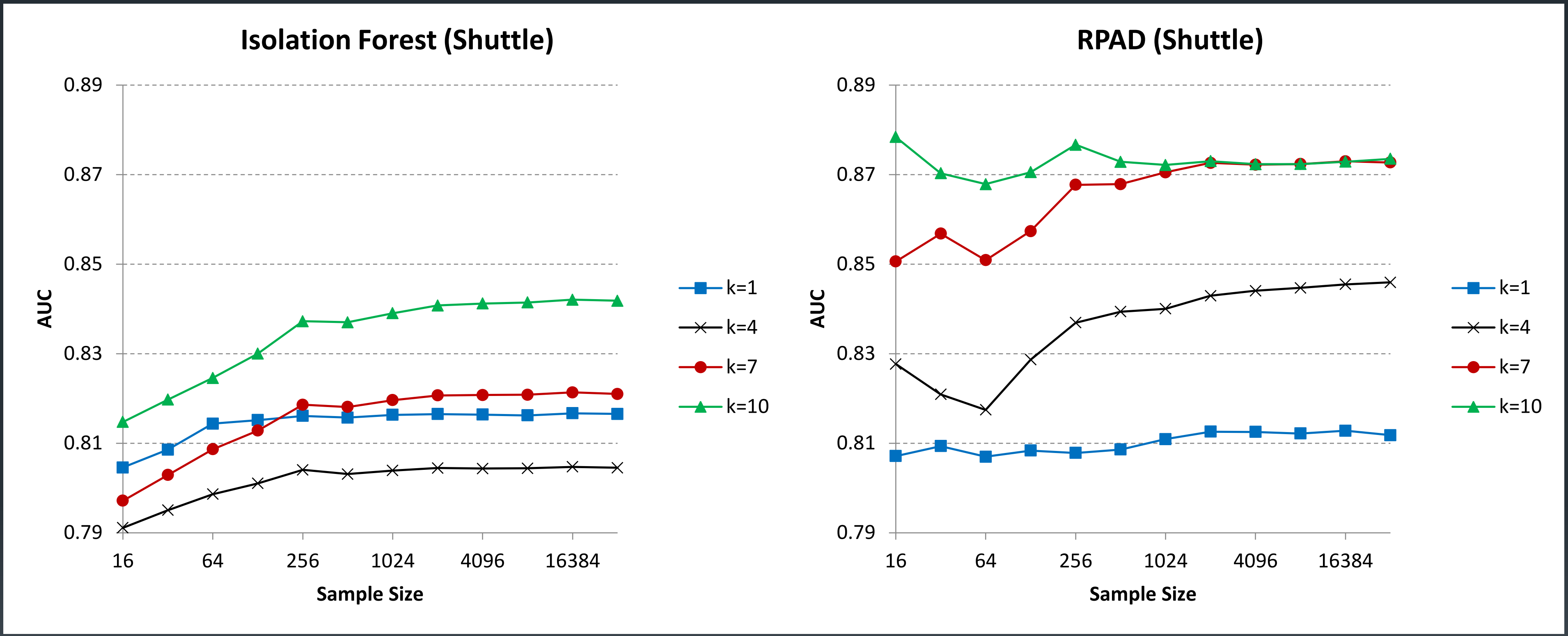
- half spaces
- axis-aligned hyper-rectangles
- stripes (equivalent to LODA's histogram bins)
- ellipsoids
- ellipsoidal shells (difference of two ellipsoidal level sets)

Isolation RPAD (aka Pattern Min)

- Grow an isolation forest
  - Each tree is only grown to depth k
  - Each leaf defines a pattern h
  - μ is the volume (Lebesgue measure)
  - Compute $\hat{f}(h)$ for each leaf
- Details
  - Grow the tree using one sample
  - Estimate $\hat{f}$ using a second sample
  - Score query point(s)



Results: Shuttle



- PatternMin is consistently better for $k > 1$

RPAD Conclusions

- The PAC-RPAD theory seems to capture the behavior of algorithms such as IFOREST
- It is easy to design practical RPAD algorithms
- Theory requires extension to handle sample-dependent pattern spaces $\mathcal{H}$

Summary

- Outlier Detection can perform unsupervised or clean anomaly detection when the relative frequency of anomalies, α is small
- Algorithm Benchmarking
  - The Isolation Forest is a robust, high-performing algorithm
  - The OCSVM and SVDD methods do not perform well on AUC and AP. Why not?
  - The other methods (ABOD, LODA, LOF, EGMM, RKDE) are very similar to each other
- Sequential Feature Explanations provide a well-defined and objectively measurable method for anomaly explanation
- Expert Feedback can be incorporated into LODA via a modified Accuracy-at-the-Top algorithm with good results
- PAC-RPAD theory may account for the rapid learning of many anomaly detection algorithms