Histogram-based Outlier Score (HBOS): A fast Unsupervised Anomaly Detection Algorithm

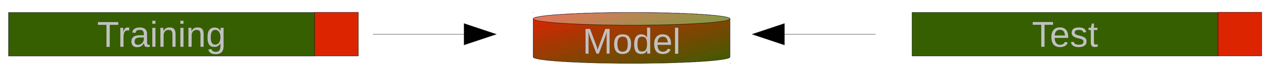
Markus Goldstein and Andreas Dengel

German Research Center for Artificial Intelligence (DFKI), Kaiserslautern

www.dfki.de {markus.goldstein,andreas.dengel}@dfki.de

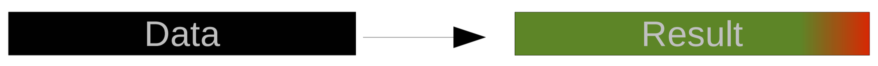
Introduction

- Anomaly detection finds **outliers** in data sets which
  - only occur very rarely in the data and
  - their features do differ from the normal instances significantly
- Three different anomaly detection setups exist [4]:
  1. Supervised anomaly detection (labeled training and test set)



  2. Semi-supervised anomaly detection (training with normal data only and labeled test set)

  3. Unsupervised anomaly detection (one data set without any labels)



- In this work, we present an **unsupervised** algorithm which **scores** instances in a given data set with respect to their outlierliness using histograms

Related Work

1. **Unsupervised anomaly detection** [4]
   - Nearest-neighbor based algorithms
     - Best performing methods today [1, 2]
     - Global k-nearest-neighbor (k-NN) [8]
     - Well known local method: Local Outlier Factor (LOF) [3]
     - Computational effort for nearest-neighbor search basically $O(n^2)$
   - Clustering based algorithms
     - Use k-means to cluster the data first
     - Compute CBLOF [5] or LDCOF [1] scores based on clustering results
     - Can be faster than k-NN methods
   - Statistical methods
     - Parametric methods, e.g. Gaussian Mixture Models (GMM)
     - Non-parametric methods, e.g. kernel-density estimation (KDE) or **histograms**
2. **Histograms in network security**
   - Histograms are used in a semi-supervised manner in network security [7]
   - Advantage: Computation is very fast $O(n)$
   - If multivariate data has to be processed, single features are scored individually and combined at the end [6]

Histogram-based Outlier Score (HBOS)

- Univariate histogram for each single feature
- Categorical data: Simple counting
- Numerical data:
  1. Static bin width with k bins having equal width
  2. Dynamic bin width with $\frac{N}{k}$ instances per bin
- Frequency (relative amount) of samples in a bin is used as density estimation
- Histograms are normalized to [0,1] for each single feature
- HBOS for each instance p is computed as a product of the inverse of the estimated density:

$$HBOS(p) = \sum_{i=0}^{d} log(\frac{1}{hist_i(p)})$$

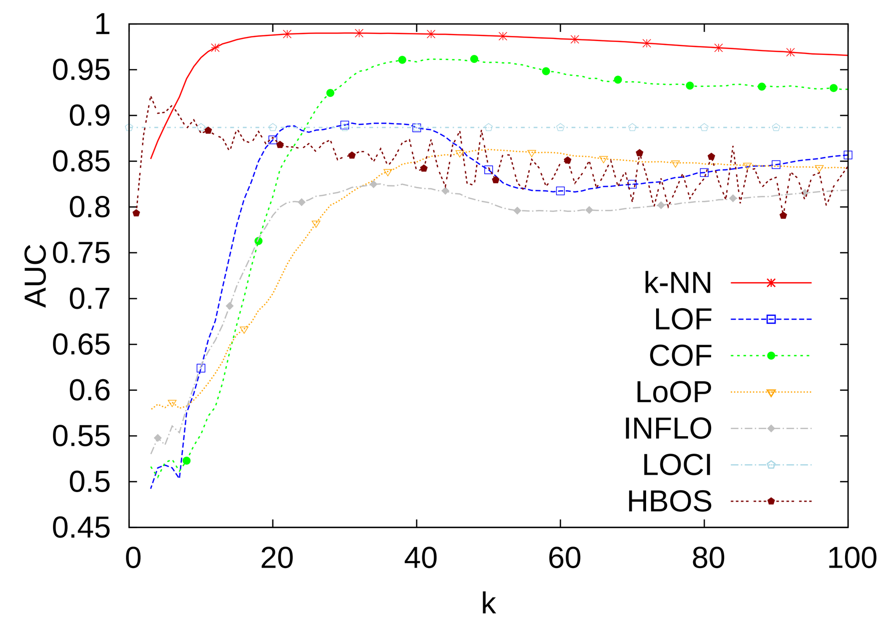
- Due to floating point precision, the product is replaced by the sum of logarithms (does not change order of scores), using $log(a \cdot b) = log(a) + log(b)$
- Assumes independence of features similar to Naive Bayes
- Different histogram techniques (categorical, static, dynamic) can be combined

Dynamic Bin Widths

- Problem using fixed bin widths: Having extreme outliers or very unbalanced distributions may lead to many empty bins (bad density estimation)
- Idea: Put the same amount of instances ($\frac{N}{k}$) into each bin (each bin has the same area)
- Bins with low density are wider but have less height
- Basically advantageous for "unbalanced" and unknown distributions
- Exception: If more than $\frac{N}{k}$ instances have exactly the same feature value, bins can contain more instances (larger area)

Evaluation and Results

- Evaluation using UCI machine learning data sets (preprocessed as in [1]):
  - Breast Cancer Wisconsin data set
  - Pen-Based Recognition of Handwritten Digits data set (global and local anomaly detection task)
- Evaluation with area under the ROC (AUC) by varying an outlier threshold



Algorithm	Breast-cancer	Pen-global	Pen-local
HBOS	**0.9910**	**0.9214**	**0.7651**
k-NN	0.9826	0.9892	0.9852
LOF	0.9916	0.8864	0.9878
Fast-LOF	0.9882	0.9050	0.9937
COF	0.9888	0.9586	0.9688
INFLO	0.9922	0.8213	0.9875
LoOP	0.9882	0.8492	0.9864
LOCI	0.9678	0.8868	—
CBLOF	0.8389	0.6808	0.7007
u-CBLOF	0.9743	0.9923	0.9767
LDCOF	0.9804	0.9897	0.9617

- Works reasonable on global anomaly detection tasks, but fails on local ones
- Speedup on the UCI data sets: 5-7 times
- Run-time on very large data set with 1,000,000 instances and 15 dimensions: LOF: 23 hrs, 46 mins; HBOS: 38 sec (static), 46 sec (dynamic)

Website and Implementation

http://madm.dfki.de/rapidminer/anomalydetection

- Part of the Anomaly Detection Extension for RapidMiner (Open Source)
- For each feature, static or dynamic approach and k can be selected

References

[1] Mennatallah Amer. Comparison of unsupervised anomaly detection techniques. Bachelor's Thesis 2011.

[2] Mennatallah Amer and Markus Goldstein. Nearest-neighbor and clustering based anomaly detection algorithms for rapidminer. In *Proc. of the 3rd RCOMM 2012*.

[3] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. Lof: identifying density-based local outliers. *SIGMOD Rec.*, 29(2):93–104, 2000.

[4] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3):1–58, 2009.

[5] Zengyou He, Xiaofei Xu, and Shengchun Deng. Discovering cluster-based local outliers. *Pattern Recognition Letters*, 24(9-10):1641 – 1650, 2003.

[6] Yoohwan Kim, Wing Cheong Lau, and et al. Packetscore: statistics-based overload control against distributed denial-of-service attacks. In *INFOCOM 2004.*, volume 4, pages 2594 – 2604.

[7] A. Kind, M.P. Stoecklin, and X. Dimitropoulos. Histogram-based traffic anomaly detection. *Network and Service Management, IEEE Transactions on*, 6(2):110 –121.

[8] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. Efficient algorithms for mining outliers from large data sets. SIGMOD '00, pages 427–438.