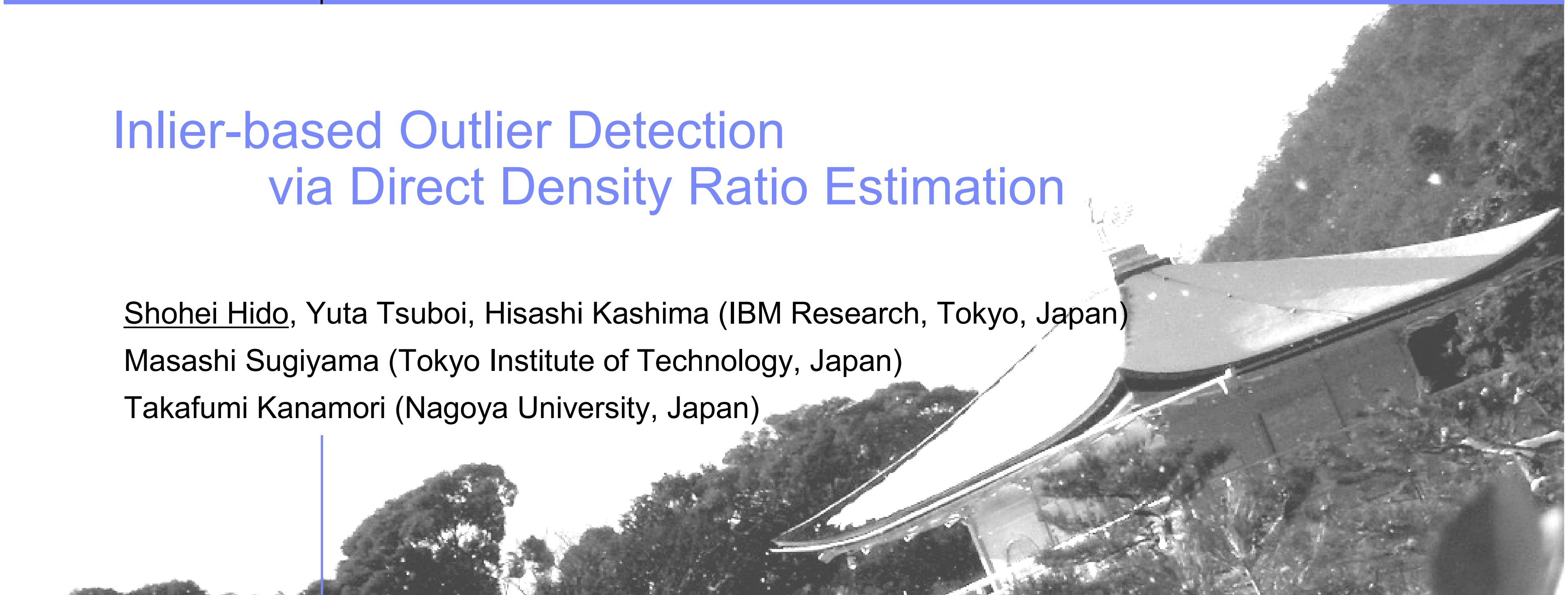
IBM Tokyo Research Laboratory

Inlier-based Outlier Detection via Direct Density Ratio Estimation

Shohei Hido, Yuta Tsuboi, Hisashi Kashima (IBM Research, Tokyo, Japan)

Masashi Sugiyama (Tokyo Institute of Technology, Japan)

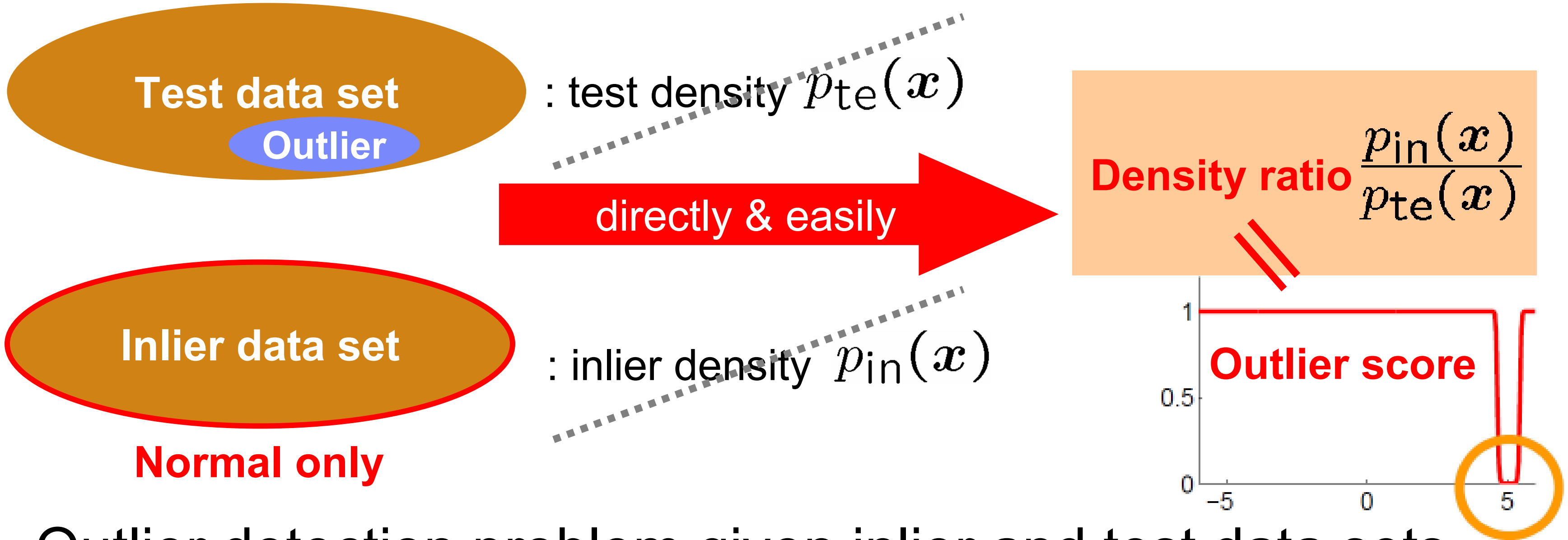
Takafumi Kanamori (Nagoya University, Japan)



IEEE ICDM 2008 | Pisa, Italy | 17/12/2008

Overview: Density ratio as outlier score

Goal: to detect outliers in a test set given normal samples



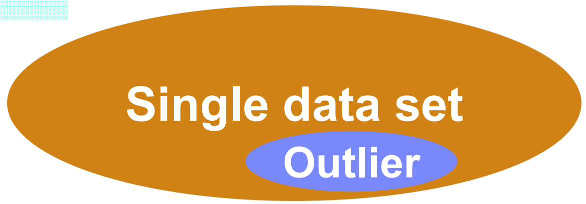
1. Outlier detection problem given inlier and test data sets
2. Direct density ratio estimation for scoring outlier-ness
3. Evaluation using benchmark and real-world data set

Outline

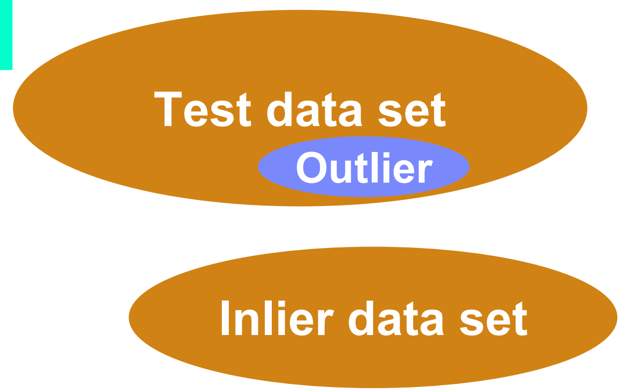
- Inlier-based outlier detection
  - Problem definition and applications
  - Density ratio as an outlier score
- Algorithms
  - Direct density ratio estimation: KLIEP & uLSIF
  - Comparison with other detection algorithms
- Experiments
  - Artificial and benchmark data sets
  - Fault prediction in hard disk systems

Motivation: Outlier detection given inlier (regular) data sets

- **Traditional outlier detection problem**
  - Given a single data set
    - Regular samples and a few outliers



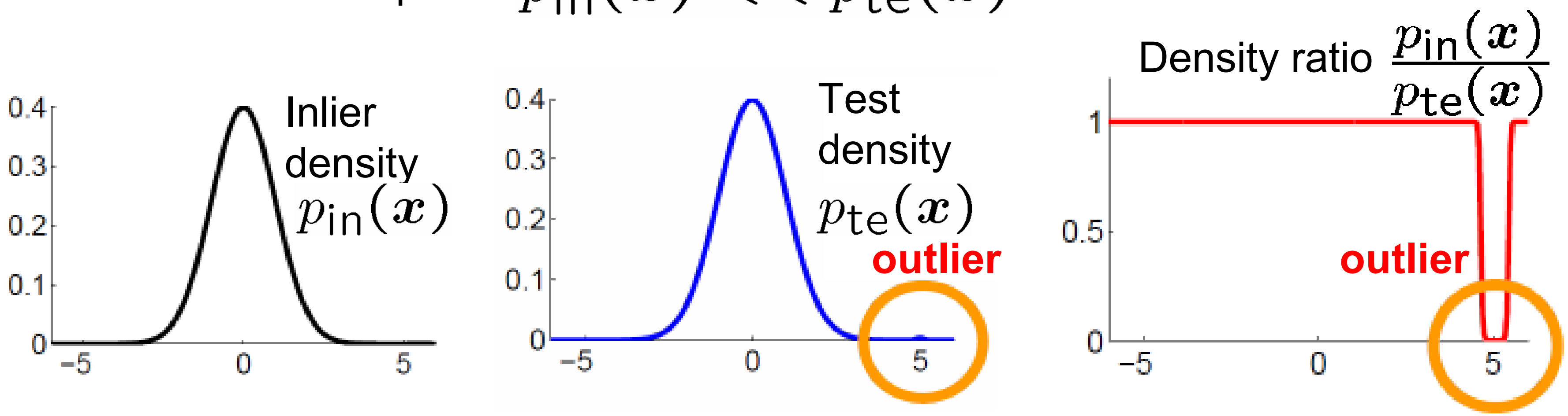
- **Inlier-based outlier detection problem**
  - Given two data sets
    1. Test data set: might include outliers
    2. Inlier data set: only regular samples
  - Real-world applications
    - Fault diagnosis: user usage data vs. controlled test data
    - New topic detection: recent documents vs. old documents



What to do for this new detection problem?

Idea: Ratio of densities can be outlier score

- Outlier score: output of outlier detection algorithms
  - Then decide outliers based on a threshold
- "Density ratio = outlier score": outliers have larger test density
  - For regular samples: $p_{\mathrm{in}}(\boldsymbol{x}) \doteqdot p_{\mathrm{te}}(\boldsymbol{x})$
  - For outlier samples: $p_{\mathrm{in}}(\boldsymbol{x}) << p_{\mathrm{te}}(\boldsymbol{x})$



Thus we estimate the density ratio, directly

Why direct density ratio estimation?

- **Reason: density estimation is hard!**
  - Ex. Kernel Density Estimator (KDE)

$$\widehat{p}(\boldsymbol{x}) = \frac{1}{n_{\mathrm{te}}(2\pi\sigma^2)^{d/2}} \sum_{k=1}^{n} \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{2\sigma^2}\right)$$

  - Curse of dimensionality: massive data samples required
- **Vapnik's principle: never solve harder sub-problem**

Inlier density $p_{\mathrm{in}}(\boldsymbol{x})$ hard

Test density $p_{\mathrm{te}}(\boldsymbol{x})$ hard

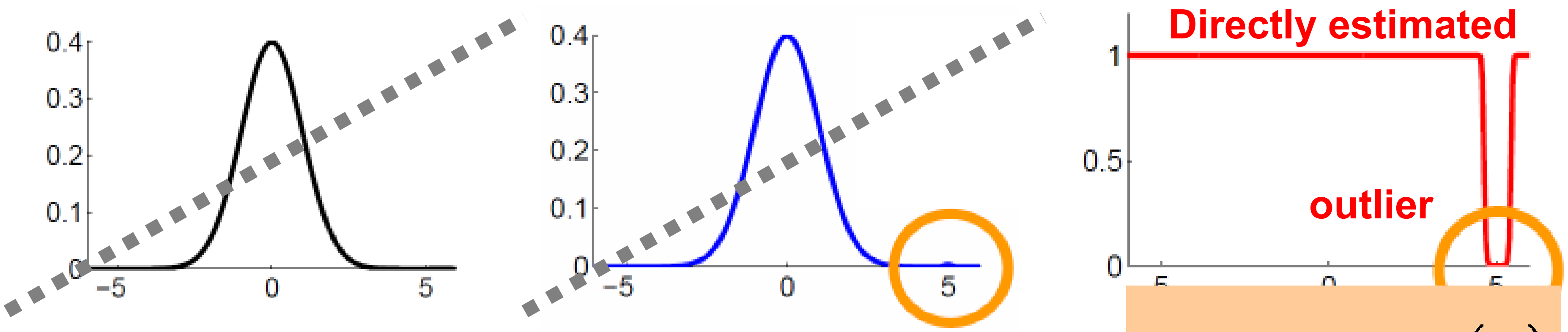
$\dfrac{p_{\mathrm{in}}(\boldsymbol{x})}{p_{\mathrm{te}}(\boldsymbol{x})}$ Density ratio

easier

Direct estimation must be easier and more accurate

Our approach:
Inlier-based outlier detection by direct density ratio estimation

- Density ratio of inlier and test data sets as outlier score
- We could apply existing direct density ratio estimation methods

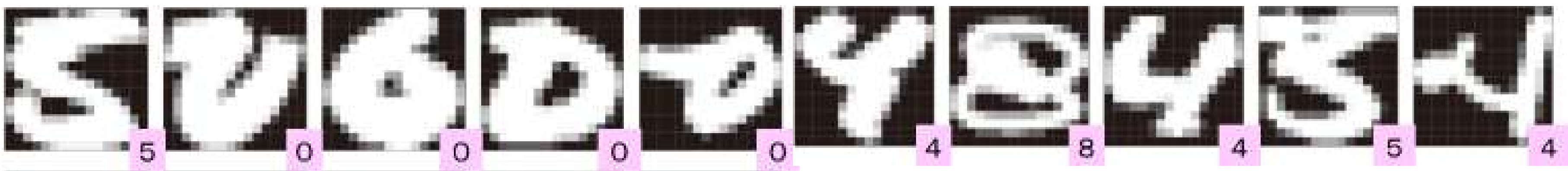


Inlier density $p_{\mathrm{in}}(\boldsymbol{x})$
hard

Test density $p_{\mathrm{te}}(\boldsymbol{x})$
hard

Density ratio $\frac{p_{\mathrm{in}}(\boldsymbol{x})}{p_{\mathrm{te}}(\boldsymbol{x})}$
easier

- Ex. Irregular digits in USPS image database



Outline

- Inlier-based outlier detection
  - Problem definition and applications
  - Density ratio as an outlier score
- Algorithms
  - Direct density ratio estimation: KLIEP & uLSIF
  - Comparison with other detection algorithms
- Experiments
  - Artificial and benchmark data sets
  - Fault prediction in hard disk systems

Problem: Minimize estimation error using linear density ratio model

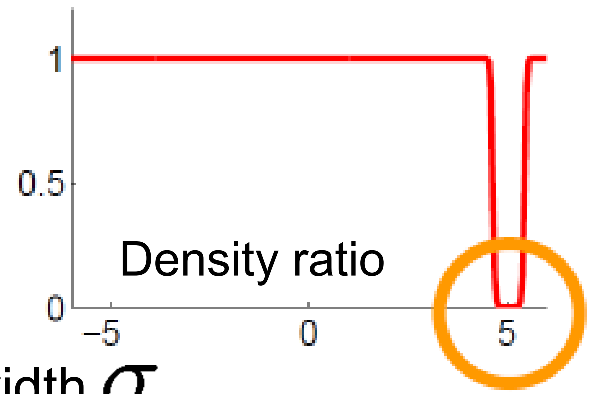
- Data sets

Inlier data set : $\{\boldsymbol{x}_j^{\mathrm{in}}\}_{j=1}^{n_{\mathrm{in}}}$

Test data set : $\{\boldsymbol{x}_i^{\mathrm{te}}\}_{i=1}^{n_{\mathrm{te}}}$

- True density ratio

$$w(\boldsymbol{x}) = \frac{p_{\mathrm{in}}(\boldsymbol{x})}{p_{\mathrm{te}}(\boldsymbol{x})}$$



- Linear density ratio model

$$\widehat{w}(\boldsymbol{x}) = \sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x})$$

- Estimation using Gaussian kernel with width σ

$$\widehat{w}(\boldsymbol{x}) = \sum_{\ell=1}^{b} \alpha_\ell K_\sigma(\boldsymbol{x}, \boldsymbol{x}_\ell^{\mathrm{in}}) \quad \text{for } \boldsymbol{x} \in \{\boldsymbol{x}_i^{\mathrm{te}}\}_{i=1}^{n_{\mathrm{te}}}$$

$$K_\sigma(\boldsymbol{x}, \boldsymbol{x}') = \exp\left\{-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right\}$$

Goal: to obtain the optimal coefficients α_ℓ

KLIEP: Kullback-Leibler Importance Estimation Procedure

Sugiyama, Nakajima, Kashima, von Bünau & Kawanabe (NIPS2007)

- Loss: Kullback-Leibler loss

$$KL[p_{\mathrm{in}}(\boldsymbol{x}) \| \widehat{p}_{\mathrm{in}}(\boldsymbol{x})] = \int_{\mathcal{D}} p_{\mathrm{in}}(\boldsymbol{x}) \log \frac{p_{\mathrm{in}}(\boldsymbol{x})}{\widehat{w}(\boldsymbol{x}) p_{\mathrm{te}}(\boldsymbol{x})} d\boldsymbol{x}.$$

- Objective: convex and not including true densities

$$\text{maximize} \ \frac{1}{n_{\mathrm{in}}} \sum_{j=1}^{n_{\mathrm{in}}} \log \left(\sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}_j^{\mathrm{in}}) \right)$$

$$\text{subject to} \ \sum_{i=1}^{n_{\mathrm{te}}} \sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}_i^{\mathrm{te}}) = n_{\mathrm{te}} \ \text{and} \ \alpha_1, \alpha_2, \ldots, \alpha_b \geq 0.$$

- Optimization: gradient ascent + constraint satisfaction (repeated)
- Advantage
  - Global optima
  - Equipped with model selection by likelihood cross validation (LCV)
  - Good estimation accuracy in high dimension

uLSIF: Unconstrained Least-Squares Importance Fitting

Kanamori, Hido & Sugiyama (NIPS2008)

- Loss: squared loss

$$\frac{1}{2}\int\left(\widehat{w}(\boldsymbol{x})-\frac{p_{\mathrm{in}}(\boldsymbol{x})}{p_{\mathrm{te}}(\boldsymbol{x})}\right)^2 p_{\mathrm{te}}(\boldsymbol{x})d\boldsymbol{x}$$

- Objective: with L2 regularization without non-negativity constraint

$$\text{minimize } \frac{1}{2}\boldsymbol{\alpha}^\top\widehat{\boldsymbol{H}}\boldsymbol{\alpha}-\widehat{\boldsymbol{h}}^\top\boldsymbol{\alpha}+\lambda\boldsymbol{\alpha}^\top\boldsymbol{\alpha}$$

$$\text{where } \widehat{\boldsymbol{H}}=\frac{1}{n_{\mathrm{te}}}\sum_{i=1}^{n_{\mathrm{te}}}\boldsymbol{\varphi}(\boldsymbol{x}_i^{\mathrm{te}})\boldsymbol{\varphi}(\boldsymbol{x}_i^{\mathrm{te}}) \text{ and } \widehat{\boldsymbol{h}}=\frac{1}{n_{\mathrm{in}}}\sum_{j=1}^{n_{\mathrm{in}}}\boldsymbol{\varphi}(\boldsymbol{x}_j^{\mathrm{in}})$$

$$\text{for } \boldsymbol{\varphi}(\boldsymbol{x})=(\varphi_1(\boldsymbol{x}),\ldots,\varphi_b(\boldsymbol{x}))^\top.$$

- Optimization: analytically solved + non-negativity satisfaction
- Advantage
  - Stability of analytical solution
  - Leave-one-out cross validation (LOOCV) at one time: much faster
    - Based on Sherman-Woodbury-Morrison formula

Conventional outlier detection algorithms: Could be used for inlier-based outlier detection

- One-class SVM (OCSVM)

  Schölkopf, Platt, Shawe-Taylor, Smola and Williamson (Neural Comp. 2001)

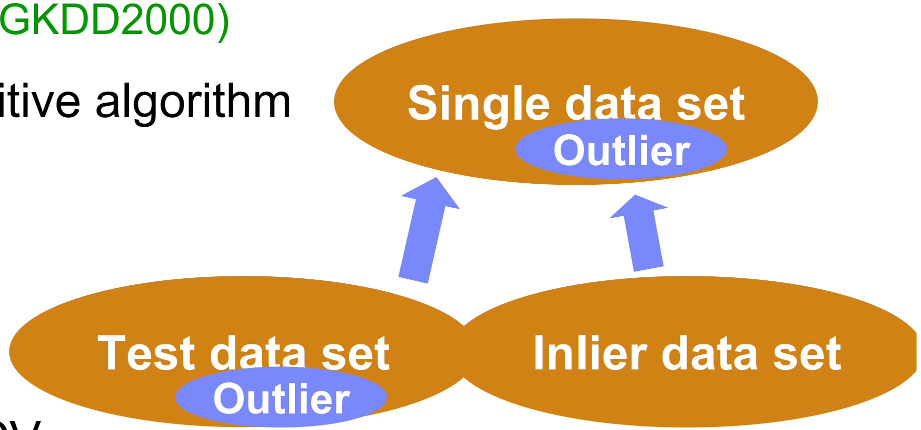
  - Modified SVM to find outlier boundary by QP solver
  - NO model selection of a few parameters at once

- Local Outlier Factor (LOF)

  Breunig, Kriegel, Ng and Sander (SIGKDD2000)

  - Nearest neighbor-based locality sensitive algorithm
  - NO model selection for parameter k

- Kernel Density Estimator (KDE)
  - Naturally applied
  - Gaussian width can be chosen via LCV



We apply them on the single merged data set

Comparison of algorithms

Our methods are qualitatively efficient

Advantage Disadvantage

		Density estimation	Model selection	Running time
Density ratio estimation	uLSIF	–	LOOCV	Short
	KLIEP	–	LCV	Normal
	LogReg	–	CV	Long
	KMM	–	–	Long
Traditional outlier detection	OCSVM	–	–	Long
	LOF	–	–	Longest
	KDE	Required	LCV	Shortest

Kernel Mean Matching (KMM): Huang, Smola, Gretton, Borgwardt and Schölkopf (NIPS2006)

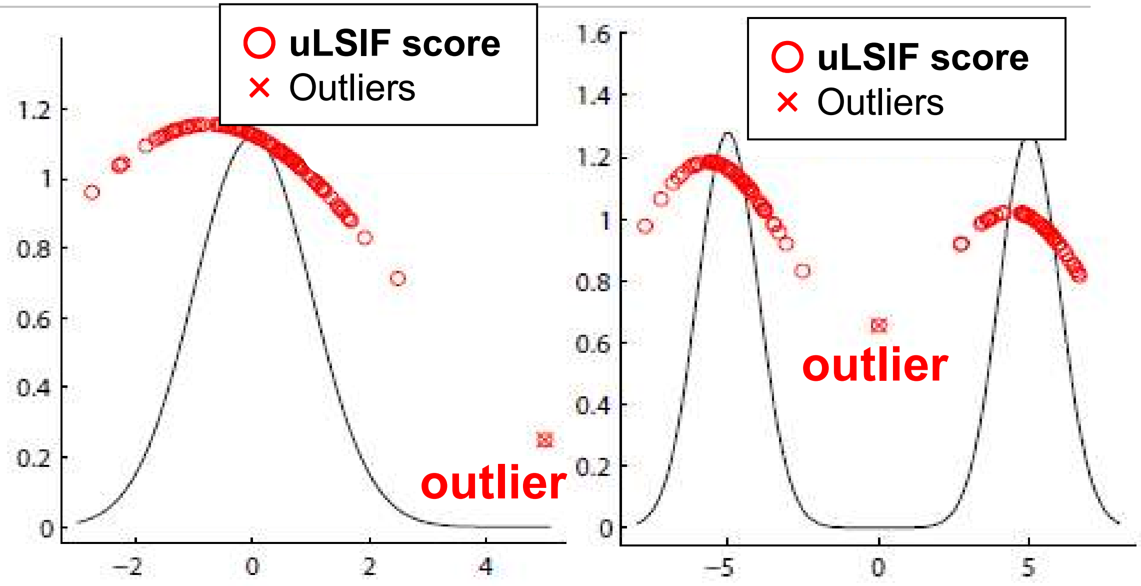
Logistic Regression method (LogReg): Bickel, Brückner and Scheffer (ICML2007)

Outline

- Inlier-based outlier detection
  - Problem definition and applications
  - Density ratio as an outlier score
- Algorithms
  - Direct density ratio estimation: KLIEP & uLSIF
  - Comparison with other detection algorithms
- Experiments
  - Artificial and benchmark data sets
  - Fault prediction in hard disk systems

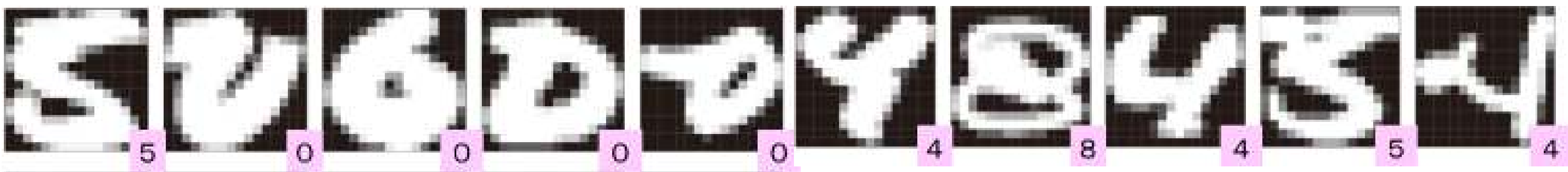
Artificial and USPS datasets: Detected outliers by our methods

- Toy example
  - Inlier: Gaussians
  - Test: Gaussians +an outlier
  - Embedded outliers got lower score



(a) $\mathcal{N}(x;0,1)$ (b) $\frac{1}{2}(\mathcal{N}(x;-5,1)+\mathcal{N}(x;5,1))$

- Hard test examples in USPS image database
  - Unclear and mislabeled samples were detected as outliers



Experimental setting: comparison with density ratio estimation and outlier detection methods

- Data set
  - 12 data set from Raetsch: converted into outlier detection problem

    Inlier set **Negative samples** Test set **Negative samples** **positive**

  - SMART data set: good hard disks under reliability test and user's failed disks

    Inlier set **Good disks** Test set **Good disks** **failed**

- Parameters
  - Model selection for Gaussian width: CV / LCV / LOOCV
  - k = {5, 30, 50} : Number of Neighbors for LOF
  - r = {0.01, 0.02, 0.05} : Changing outlier population
  - b = 100 : Number of Gaussian centers is fixed
- Evaluation metric:
  - AUC (Area under ROC curves) value
  - Computation time (normalized with that of uLSIF)

Raetsch data sets:
Our methods are accurate and faster

Data set	Outliers	uLSIF	KLIEP	LogReg	KMM	OSVM	LOF (k=5)	LOF (k=30)	LOF (k=50)	KDE
banana	0.01	0.851	0.815	0.433	0.578	0.360	0.838	0.915	0.919	0.934
breast cancer	0.01	0.463	0.480	0.616	0.576	0.508	0.546	0.488	0.463	0.400
diabetes	0.01	0.558	0.615	0.595	0.574	0.563	0.513	0.403	0.390	0.425
heart	0.01	0.659	0.647	0.788	0.623	0.681	0.407	0.659	0.739	0.638
satimage	0.01	0.812	0.828	0.616	0.813	0.540	0.909	0.930	0.896	0.916
waveform	0.01	0.890	0.881	0.216	0.477	0.861	0.724	0.887	0.889	0.861
...										
Average		0.661	0.685	0.509	0.608	0.596	0.594	0.629	0.622	0.623
Comp. time		1	11.7	5.62	751	12.4	85.5			8.69

- Performance depends on each data set
- uLSIF is the fastest and KLIEP is the most accurate

SMART data sets: uLSIF worked well for the real-world application

Window size	Outliers	uLSIF	KLIEP	LogReg	KMM	OSVM	LOF (k=5)	LOF (k=30)	LOF (k=50)	KDE
5	0.01	0.894	0.842	0.851	0.822	0.919	0.854	0.937	0.933	0.918
	0.02	0.870	0.810	0.862	0.813	0.896	0.850	0.934	0.928	0.892
	0.05	0.885	0.858	0.888	0.849	0.864	0.789	0.911	0.923	0.883
10	0.01	0.868	0.805	0.827	0.889	0.812	0.880	0.925	0.920	0.557
	0.02	0.879	0.845	0.852	0.894	0.785	0.860	0.919	0.917	0.546
	0.05	0.889	0.857	0.856	0.898	0.783	0.849	0.915	0.916	0.619
...										
Average		0.881	0.836	0.856	0.861	0.843	0.847	0.924	0.923	0.736
Comp. time		1.00	1.07	3.11	4.36	26.98	65.31			2.19

Our methods are practically effective

Conclusion

- ✓ Statistical approach for inlier-based outlier detection
- ✓ Applying Direct density ratio estimation
  - ✓ KLIEP and uLSIF
  - ✓ Model selection capability is the major advantage
- ✓ Evaluation using benchmark and real-world data set
  - ✓ KLIEP and uLSIF works much faster
  - ✓ The performances are competitively accurate