Anomaly Detection

Jean-Yves Tourneret[(1)] et Axel Carlier [(2)]

(1) University of Toulouse, ENSEEIHT-IRIT-TéSA, jyt@n7.fr
(2) University of Toulouse, ENSEEIHT-IRIT, Axel.Carlier@toulouse-inp.fr

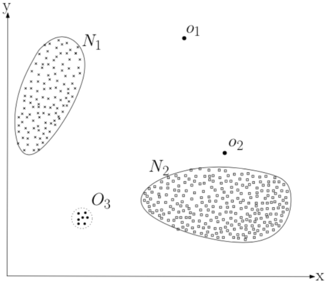
December 2024

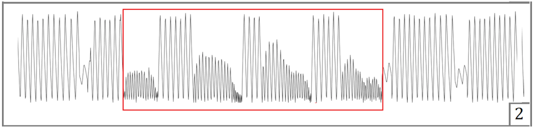
Summary

Anomaly detection

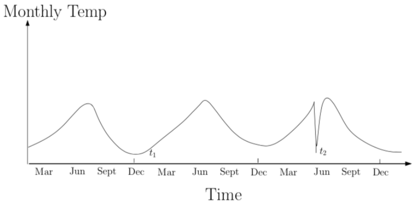
- Classes of anomalies
- Algorithms
  - Distance-based algorithms
    - LoOF and LOOP
    - Discords
  - Domain-based algorithms
    - One-Class SVM
  - Isolation Forests
  - Reconstruction-based algorithms
    - Subspace-based methods
    - Neural network-based approaches
  - Online anomaly detection

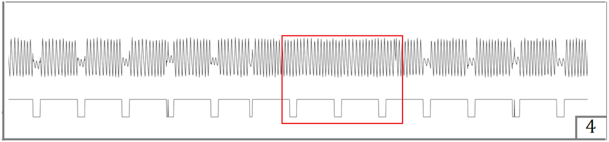
Ponctual Anomalies



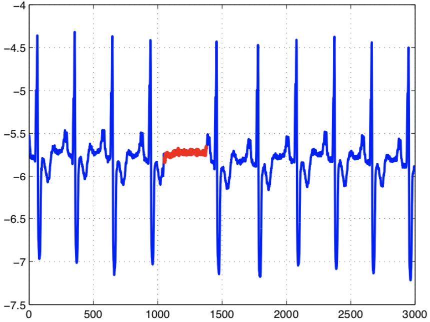


Contextual Anomalies





Collective Anomalies



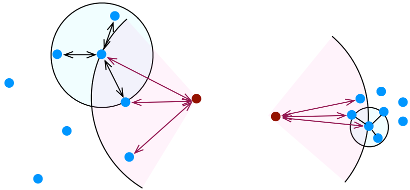
Summary

Anomaly detection

- Classes of anomalies
- Algorithms
  - Distance-based algorithms
    - LoOF and LOOP
    - Discords
  - Domain-based algorithms
    - One-Class SVM
  - Isolation Forest
  - Reconstruction-based algorithms
    - Subspace-based methods
    - Neural network-based approaches
  - Online anomaly detection

Local Outlier Factor (LOF) [Breunig, 2000]

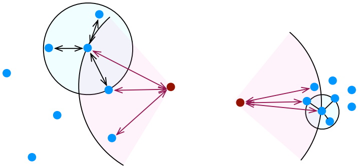
- General principle of k-NN methods: anomalies are far from nominal data and in areas where there are few nominal data



- LOF is based on a "local density" in the neighborhood of each point (with a specific distance referred to as "local reachability distance")

$$\mu(\boldsymbol{x}_i) = \left(\frac{1}{|\mathcal{N}_k(\boldsymbol{x}_i)|} \sum_{\boldsymbol{x}_j \in \mathcal{N}_k(\boldsymbol{x}_i)} d_k(\boldsymbol{x}_i, \boldsymbol{x}_j) \right)^{-1}, \quad \mathcal{N}_k(\boldsymbol{x}_i)\text{: } k\text{-NN of } \boldsymbol{x}_i$$

Local Outlier Factor



- If the local density of a test point is close to the density of its neighbors, this point is declared as “normal”.

Local Outlier Factor

- Definition

$$\text{LOF}_k(\boldsymbol{x}_i) = \frac{\frac{1}{|\mathcal{N}_k(\boldsymbol{x}_i)|}\sum_{\boldsymbol{x}_j \in \mathcal{N}_k(\boldsymbol{x}_i)} \mu(\boldsymbol{x}_j)}{\mu(\boldsymbol{x}_i)}.$$

  If $\boldsymbol{x}_i$ is in a homogeneous area (normal point) $\text{LOF}_k(\boldsymbol{x}_i) \approx 1$, else $\text{LOF}_k(\boldsymbol{x}_i) >> 1$ (density of the neighbors of $\boldsymbol{x}_i$ larger than density of $\boldsymbol{x}_i$).

- Reachability distance between p and o

  In order to reduce the fluctuation of $d(p, o)$ when p is close to o, one can use the reachability distance

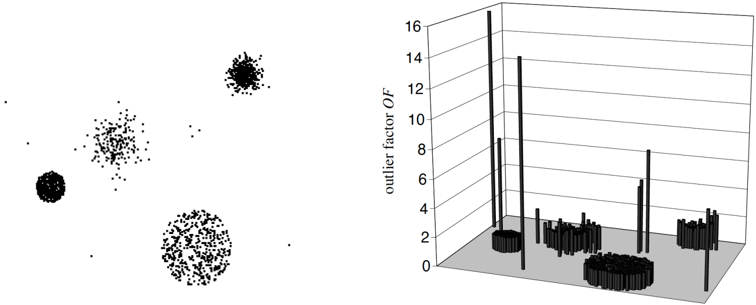
$$\text{rd}_k(p, o) = \max\{d_k(p, o), d(p, o)\}$$

  - If p is far from o, then $\text{rd}_k(p, o) = d(p, o)$
  - If p is close to o, $\text{rd}_k(p, o)$ is the distance between p and the kth nearest neighbor of o

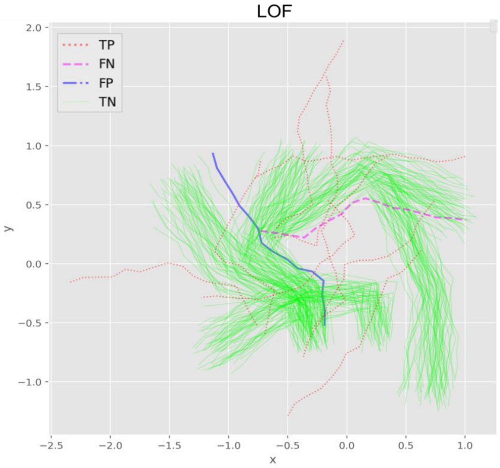
Example from [Breunig, 2000]

LOF values for $k = 30$ and $n = 1700$

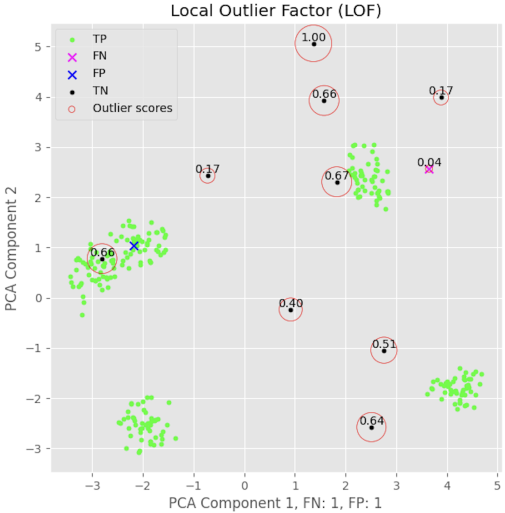
One low density Gaussian cluster of 200 objects and three large clusters of 500 objects each.



LOF for Maritime Surveillance ($k = 9$, Contamination $= 10/260$)



Inverse LOF for Maritime Surveillance ($k = 9$, Contamination $= 10/260$)



Local Outlier Probabilities (LoOP) [Kriegel, 2009]

- LoOP reformulates LOF in a probabilistic context by normalizing $\text{LOF}_k(\boldsymbol{x}_i)$ and deriving an anomaly score $\in]0, 1[$ for each vector $\boldsymbol{x}_i$:

$$\text{LoOP}_k(\boldsymbol{x}_i) : \text{probability that } \boldsymbol{x}_i \text{ is an anomaly}$$

- Parameters of LoOP
  - Number of nearest neighbors k: to be determined by cross validation.
  - One significance parameter λ ensuring that a point o is an outlier for S if

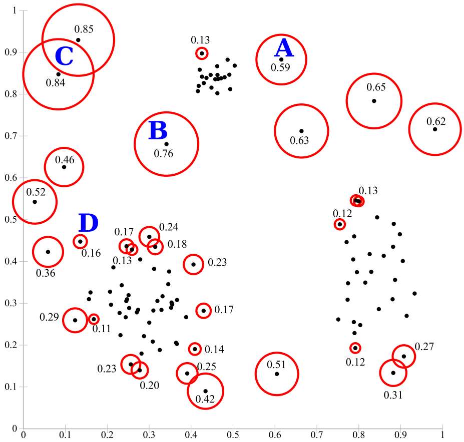
$$P[0 < d(o, s) < \lambda\sigma(o, S)] < \phi, \forall s \in S.$$

where $\sigma(o, S)$ is a kind of average distance between o and the elements of S:

$$\sigma(o, S) = \sqrt{\frac{\sum_{s \in S} d^2(o, s)}{|S|}}.$$

As examples, assuming that $\frac{d(o,s}{\sigma(o,s)}$ is distributed according to a half $\mathcal{N}(0, 1)$ distribution, we obtain $\lambda = 3$ if $\phi = 99.7\%$ and $\lambda = 2$ if $\phi = 95\%$.

Examples of LoOPs ($k = 20, \lambda = 3$)



Summary

Anomaly detection

- Classes of anomalies
- Algorithms
  - Distance-based algorithms
    - LoOF and LOOP
    - Discords
  - Domain-based algorithms
    - One-Class SVM
  - Isolation Forest
  - Reconstruction-based algorithms
    - Subspace-based methods
    - Neural network-based approaches
  - Online anomaly detection

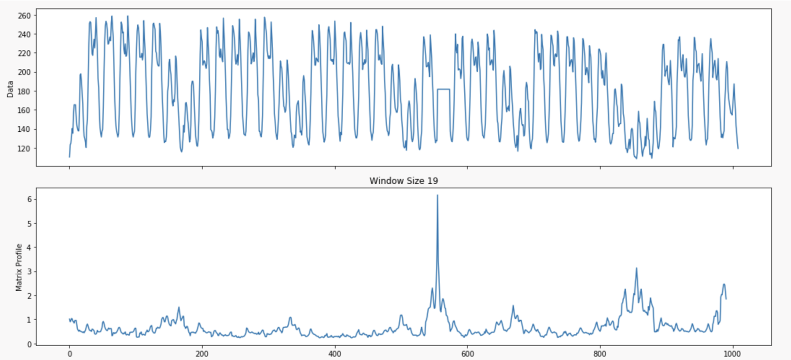
Discords [Keogh, 2005]

- Non-Self Match: M is a non-self match of C at distance of $\text{dist}(M, C)$ if M of length n begins at p, C of length n begins at q and $|p - q| \geq n$.

  a b c a b c a b c a b c X X X a b c a b c a b a c a b c

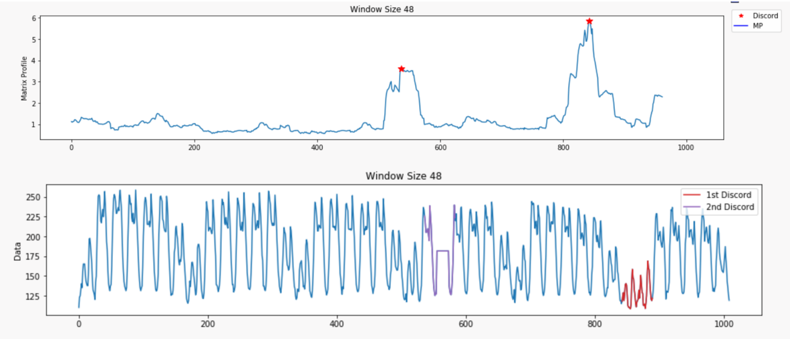
- Time Series Discord : Given a time series T, the subsequence D of length n beginning at position p is called the discord of T, if D has the largest distance to its nearest non-self match.
- kth Time Series Discord : Given a time series T, the subsequence D of length n beginning at position p is called the kth-discord of T if D has the kth largest distance to its nearest non-self match.

One discord



Discord for the hourly power electrical demand in an Italian city during 42 days (1008 hours) - $n = 19$ hours (anomaly size), $k = 1$ (https://matrixprofile.org/posts/what-are-time-series-discords/).

Two discords



Discord for the hourly power electrical demand in an Italian city during 42 days (1008 hours) - $n = 48$ hours (anomalies that last 2 days), $k = 2$ (https://matrixprofile.org/posts/what-are-time-series-discords/).

Summary

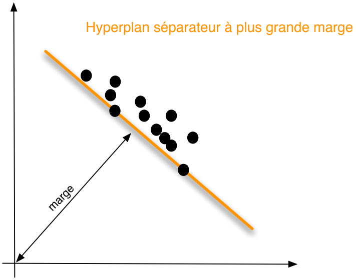
Anomaly detection

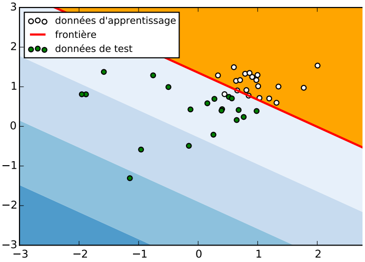
- Classes of anomalies
- Algorithms
  - Distance-based algorithms
    - LoOF and LOOP
    - Discords
  - Domain-based algorithms
    - One-Class SVM
  - Isolation Forest
  - Reconstruction-based algorithms
    - Subspace-based methods
    - Neural network-based approaches
  - Online anomaly detection

Linear One-Class-SVM method

- Find the hyperplane separating the training data $\mathcal{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$ from the origin and located as far as possible from the origin
- Distance between a point $\boldsymbol{x} = (x, y)^T$ and a straight line $\mathcal{D}$ of equation $\alpha x + \beta y - \rho = 0$

$$d(\boldsymbol{x}, \mathcal{D}) = \frac{|\alpha x + \beta y - \rho|}{\sqrt{\alpha^2 + \beta^2}} = \frac{|\boldsymbol{w}^T \boldsymbol{x} - \rho|}{\|\boldsymbol{w}\|}$$





Linear One-Class-SVM method

- By noting that the margin is $d(\mathbf{0}, \mathcal{D}) = \frac{\rho}{\|\boldsymbol{w}\|}$, we can solve the following optimization problem ("Soft-margin" SVM classifier)

$$\text{minimize } \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}\xi_i$$
$$\text{with the constraints } \boldsymbol{w}^T\boldsymbol{x}_i \geq 1 - \xi_i,\ \ \xi_i \geq 0, \forall i$$

or the $\nu-$SVM formulation

$$\text{Minimize } \frac{1}{2}\|\boldsymbol{w}\|^2 + \frac{1}{n\nu}\sum_{i=1}^{n}\xi_i - \rho$$
$$\text{with the constraints } \boldsymbol{w}^T\boldsymbol{x}_i \geq \rho - \xi_i,\ \ \xi_i \geq 0, \forall i, \rho \geq 0$$

ensuring that the percentage of vectors violating the constraint $\boldsymbol{w}^T\boldsymbol{x}_i - \rho \geq 0$ is upper-bounded by ν and that the fraction of support vectors is lower bounded by ν.

Optimization

Kühn and Tucker multipliers

For a convex optimization problem (convex function $f(\boldsymbol{x})$ to optimize and convex constraints $G_i(\boldsymbol{x}) \leq 0$), an optimality condition is the existence of parameters $\alpha_i \geq 0$ such that the Lagrangian derivative is zero, i.e.,

$$L'(\boldsymbol{x}) = f'(\boldsymbol{x}) + \sum_{i=1}^{n} \alpha_i G_i'(\boldsymbol{x}) = 0$$

with $\alpha_i = 0$ if $G_i(\boldsymbol{x}) < 0$ (i.e., $\alpha_i G_i(\boldsymbol{x}) = 0$).

Optimization

Lagrangian

$$L\left(\widetilde{\boldsymbol{w}}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \rho\right) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{n\nu}\sum_{i=1}^{n}\xi_i - \rho - \sum_{i=1}^{n}\alpha_i\left(\boldsymbol{w}^T\boldsymbol{x}_i - \rho + \xi_i\right) - \sum_{i=1}^{n}\beta_i\xi_i$$

Set to zero the partial derivatives of L with respect to the primal variables $\boldsymbol{w}$, $\boldsymbol{\xi}$ and ρ to zero yields

$$\boldsymbol{w} = \sum_{i=1}^{n}\alpha_i\boldsymbol{x}_i, \sum_{i=1}^{n}\alpha_i = 1 \text{ and } \alpha_i = \frac{1}{n\nu} - \beta_i \leq \frac{1}{n\nu}, \forall i$$

Remark on support vectors

- Since $\alpha_i = \frac{1}{n\nu} - \beta_i$, when $\beta_i = 0$, one has $\alpha_i = \frac{1}{n\nu}$ and $\boldsymbol{x}_i$ is a support vector
- When $\beta_i > 0$, one has $\xi_i = 0$. If $\alpha_i > 0$, one has $\boldsymbol{w}^T\boldsymbol{x}_i - \rho = 0$, and $\boldsymbol{x}_i$ is also a support vector

Dual problem

Solve $L'(\boldsymbol{x}) = 0$

$$\boldsymbol{w} = \sum_{\text{Support vectors}} \alpha_i \boldsymbol{x}_i = \boldsymbol{x}^T \boldsymbol{\alpha} \tag{1}$$

with $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_n)^T$, $\boldsymbol{x} = (x_1, ..., x_n)^T$ and

$$\begin{cases} \alpha_i = 0 \text{ if the constraint is a strict inequality} \\ \alpha_i > 0 \text{ if the constraint is an equality} \end{cases}$$

After replacing the expression of $\boldsymbol{w}$ in the Lagrangian, we obtain

$$U(\boldsymbol{\alpha}) = -\frac{1}{2} \boldsymbol{\alpha}^T \left(\boldsymbol{x} \boldsymbol{x}^T \right) \boldsymbol{\alpha}$$

that has to be maximized in the domain defined by $\sum_{i=1}^n \alpha_i = 1$ and $0 \leq \alpha_i \leq \frac{1}{n\nu}$.

Remarks

Simple optimization problem

- Quadratic (hence convex) function to optimize and linear constraints
- Expression of ρ: the constraints are equalities when $\alpha_i > 0$ and $\beta_i > 0$:

$$\rho = \boldsymbol{w}^T \boldsymbol{x}_i = \sum_{j=1}^{n} \alpha_j \boldsymbol{x}_j^T \boldsymbol{x}_i.$$

- Classification rule for a vector $\boldsymbol{x}$

$$f(\boldsymbol{x}) = \text{sign} \left(\sum_{x_i \text{ support vectors}} \alpha_i \boldsymbol{x}_i^T \boldsymbol{x} - \rho \right)$$

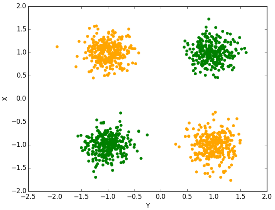
  where the summation is reduced to the support vectors.
- ν is a lower bound for the fraction of support vectors and an upper bound for the number of vectors lying outside the separating hyperplane
- Generalization to nonlinear separating curves using kernels straightforward

Non-linear SVM methods: example 1

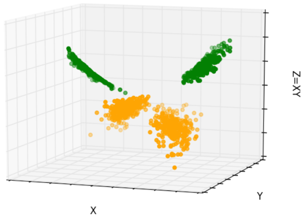
- Two classes centered around $\{(1,1)^\top, (-1,-1)^\top\}$ and $\{(1,-1)^\top, (-1,1)^\top\}$.
- Training vectors are transformed using the application ϕ

$$\begin{array}{rccc} \phi : & \mathbb{R}^2 & \longrightarrow & \mathbb{R}^3 \\ & \boldsymbol{x}_i = (x_{i,1}\ ,\ x_{i,2})^\top & \longmapsto & \phi(\boldsymbol{x}_i) = (x_{i,1}\ ,\ x_{i,2}\ ,\ x_{i,1}x_{i,2})^\top \end{array}$$

- A linear separator $\boldsymbol{w} = (0,0,1)^\top$ in the transformed space can separate the data from the two classes



(c) Original data $\boldsymbol{x}_i$ (Class #1: orange, Classe #2: green).



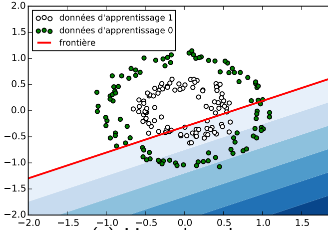
(d) Transformed data $\phi(\cdot)$

Non-linear SVM methods: example 2

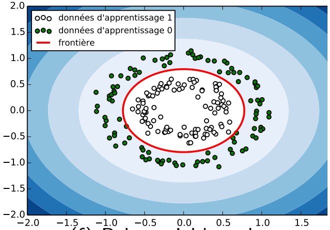
- Two classes defined by two different rings
- Polynomial transformation ϕ

$$\begin{array}{rcl} \phi : \quad \mathbb{R}^2 & \longrightarrow & \mathbb{R}^3 \\ \boldsymbol{x}_i = (x_{i,1} \,,\, x_{i,2})^\top & \longmapsto & \phi(\boldsymbol{x}_i) = (x_{i,1}^2 \,,\, x_{i,2}^2 \,,\, \sqrt{2}\, x_{i,1} x_{i,2})^\top \end{array}$$

- A linear separator $\boldsymbol{w} = (1, 1, 0)^\top$ in the transformed space corresponds to a "circular" separation in the original space.



(e) Linear kernel



(f) Polynomial kernel

Non-linear one-class SVM methods

- For two data points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, we have

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle^2.$$

  The one-class SVM only needs scalar products between the vectors $\boldsymbol{x}_i$ to be computed!

- Transposition in the ϕ domain by replacing the scalar product by a kernel

$$\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle \quad \longrightarrow \quad \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$$

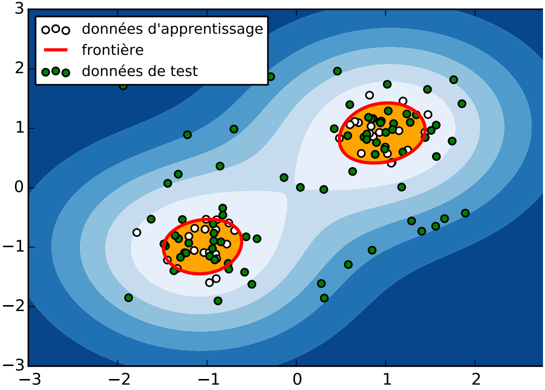
  Thus, the transformed vectors $\phi(\boldsymbol{x}_i)$ and $\phi(\boldsymbol{x}_j)$ do not need to be computed.

- Gaussian kernel

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma^2}\right).$$

  For this example, one can show that the space spanned by $\phi(\boldsymbol{x})$ has infinite dimension.

Non-linear one-class SVM methods



Parameters for the one-class SVM method

Decision rule

$$f(x) = \text{signe}\left(\sum_{i=1}^{N} \alpha_i \kappa(\boldsymbol{x}_i, \boldsymbol{x}) - \rho\right)$$

For the Gaussian kernel

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\gamma \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2\right). \tag{2}$$

Effect of the different parameters

- γ is related with the regularity of the separating curve
- ν allows the the percentage of outliers from the nominal class (located outside the separating curve) to be adjusted

Hyperparameter estimation

Hyperparameter ν

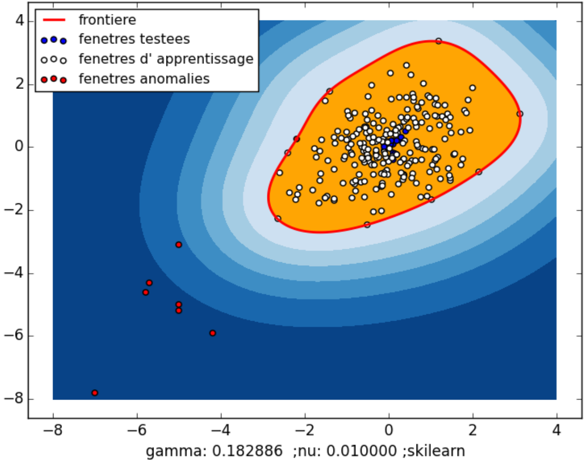
- Expert or cross validation

Hyperparameter γ

- Inverse of the number of descriptors (very adhoc)
- Cross validation
- "Trick (Jaakkola, Aggarwal, ...)": $\gamma = \frac{1}{2\sigma^2}$ with σ the median of the distances between nominal data
- More sophisticated methods are available in the literature

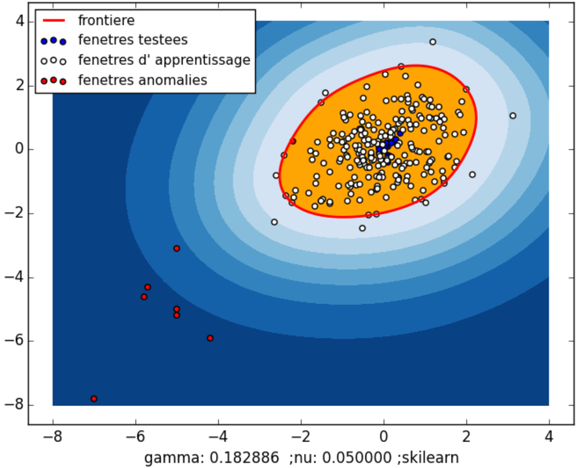
Effect of parameter ν ($\gamma = 0.18$)

$\nu = 0.01$



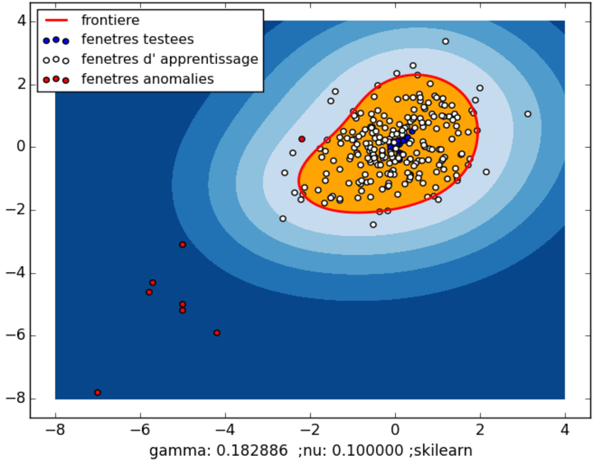
Effect of parameter ν ($\gamma = 0.18$)

$\nu = 0.05$



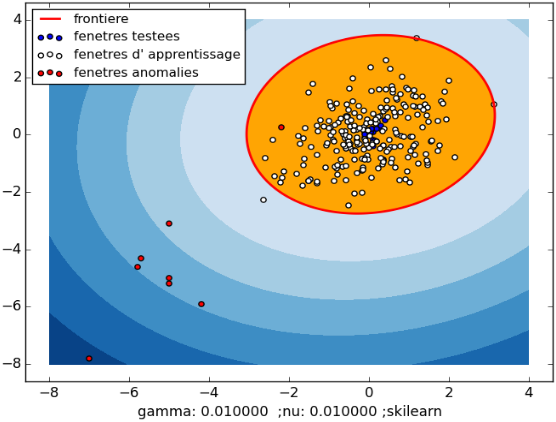
Effect of parameter ν ($\gamma = 0.18$)

$\nu = 0.1$



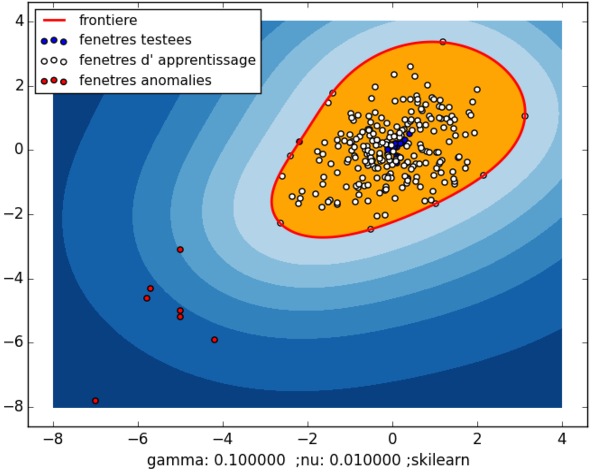
Effect of parameter γ ($\nu = 0.01$)

$\gamma = 0.01$



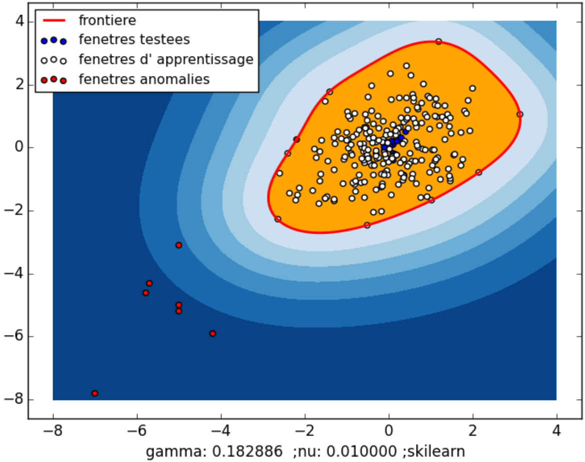
Effect of parameter γ ($\nu = 0.01$)

$\gamma = 0.1$



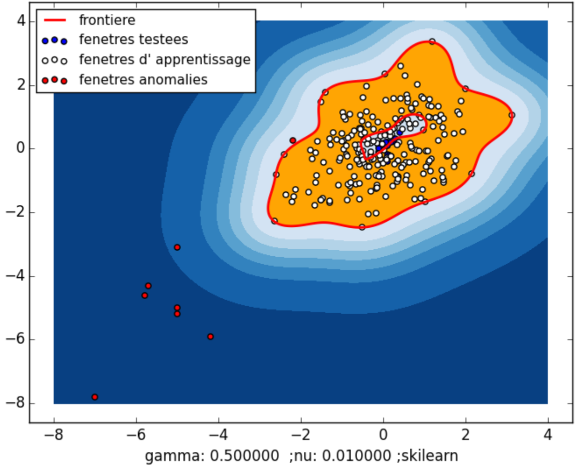
Effect of parameter γ ($\nu = 0.01$)

$\gamma = 0.18$ (Jaakkola)



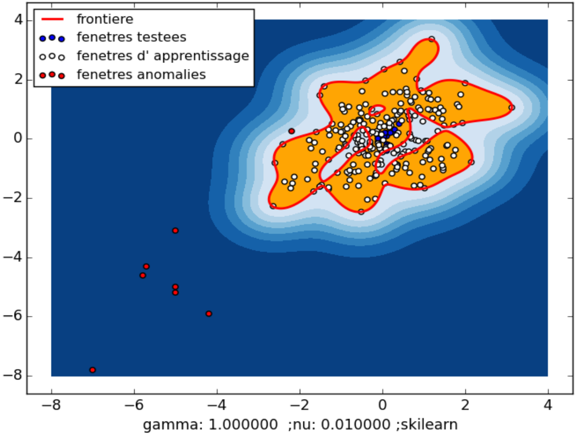
Effect of parameter γ ($\nu = 0.01$)

$\gamma = 0.5$

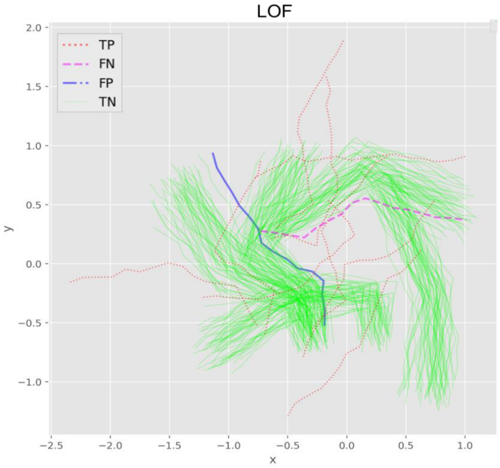


Effect of parameter γ ($\nu = 0.01$)

$\gamma = 1$

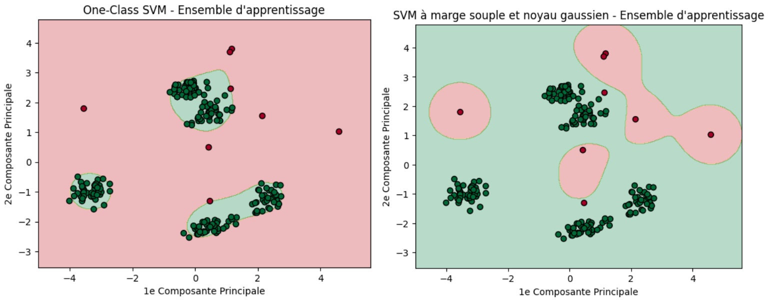


Detection of Abnormal Trajectories for Maritime Surveillance



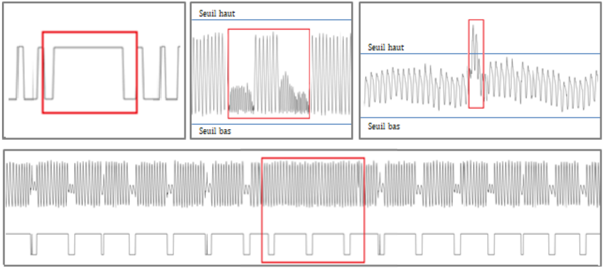
One-Class SVM versus SVM

- Left figure: one-class SVM with $\nu = 0.1$
- Right figure: supervised SVM with Gaussian kernel ($\gamma = 1$ and $C = 1$)



Application to the analysis of telemetry

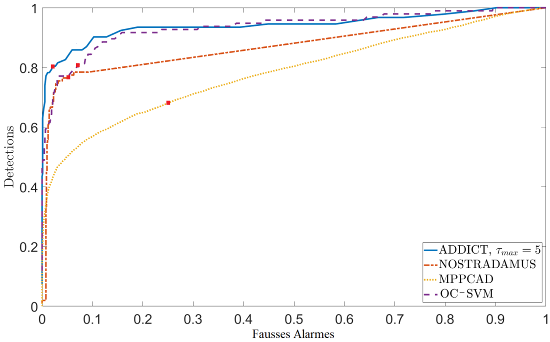
Thesis of B. Pilastre (Nov. 2020)



- Thousands of telemetry signals
- Discrete and continuous data
- Univariate and multivariate anomalies
- The out of limit (OOL) rule is simple but not efficient!

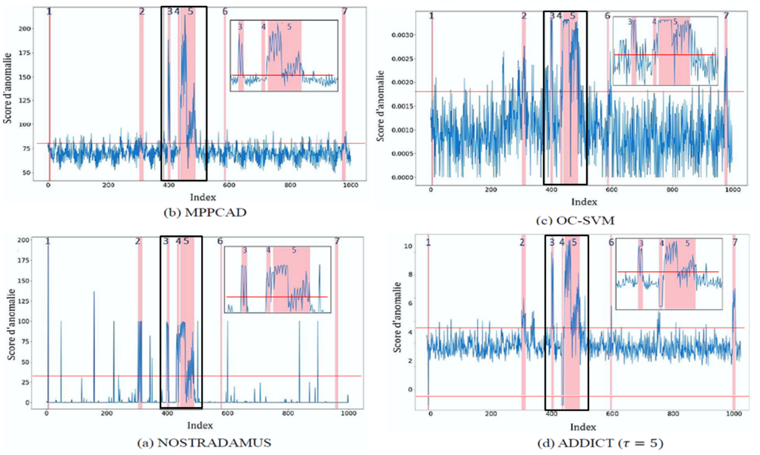
Application to the analysis of telemetry

Receiver operational characteristics



Method	Threshold	P_{D}	P_{FA}
OC-SVM	0.0018	80.85%	7%
MPPCAD	79.6	80%	13%
NOSTRADAMUS	29	77.26%	6%
ADDICT ($\tau_{\max} = 5$)	4.2	80%	3%

Detected anomalies



(b) MPPCAD

(c) OC-SVM

(a) NOSTRADAMUS

(d) ADDICT ($\tau = 5$)

Generalization to a semi-supervised scenario

Introduction of a user feedback

- Semi-supervised context: unlabelled data $\mathcal{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$, labelled normal data $\mathcal{Y} = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_n\}$ and labelled anomalies $\mathcal{Z} = \{\boldsymbol{z}_1, ..., \boldsymbol{z}_n\}$ (e.g., resulting from user feedback)
- One-class SVM with user feedback

$$\arg\min_{\boldsymbol{w},\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C_1 \sum_{i=1}^{n_1} \xi_i + C_2 \sum_{l=1}^{n_2} \zeta_l + C_3 \sum_{p=1}^{n_3} \tau_p$$

$$\text{s.t. } \boldsymbol{w}^T \Phi(\boldsymbol{x}_i) \geq 1 - \xi_i \text{ and } \xi_i \geq 0 \quad \text{unlabeled data}$$

$$\boldsymbol{w}^T \Phi(\boldsymbol{y}_l) \geq 1 - \zeta_l \text{ and } \zeta_l \geq 0 \quad \text{labeled normal}$$

$$\boldsymbol{w}^T \Phi(\boldsymbol{z}_p) \leq 1 + \tau_p \text{ and } \tau_p \geq 0 \quad \text{labeled anomalies}$$

Generalization to a semi-supervised scenario

Introduction of a user feedback

- Semi-supervised context: unlabelled data $\mathcal{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$, labelled normal data $\mathcal{Y} = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_n\}$ and labelled anomalies $\mathcal{Z} = \{\boldsymbol{z}_1, ..., \boldsymbol{z}_n\}$ (e.g., resulting from user feedback)
- One-class SVM with user feedback

$$\arg\min_{\boldsymbol{w},\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C_1 \sum_{i=1}^{n_1} \xi_i + C_2 \sum_{l=1}^{n_2} \zeta_l + C_3 \sum_{p=1}^{n_3} \tau_p$$

$$\text{s.t. } \boldsymbol{w}^T \Phi(\boldsymbol{x}_i) \geq 1 - \xi_i \text{ and } \xi_i \geq 0 \quad \text{unlabeled data}$$

$$\boldsymbol{w}^T \Phi(\boldsymbol{y}_l) \geq 1 - \zeta_l \text{ and } \zeta_l \geq 0 \quad \text{labeled normal}$$

$$\boldsymbol{w}^T \Phi(\boldsymbol{z}_p) \leq 1 + \tau_p \text{ and } \tau_p \geq 0 \quad \text{labeled anomalies}$$

Generalization to a semi-supervised scenario

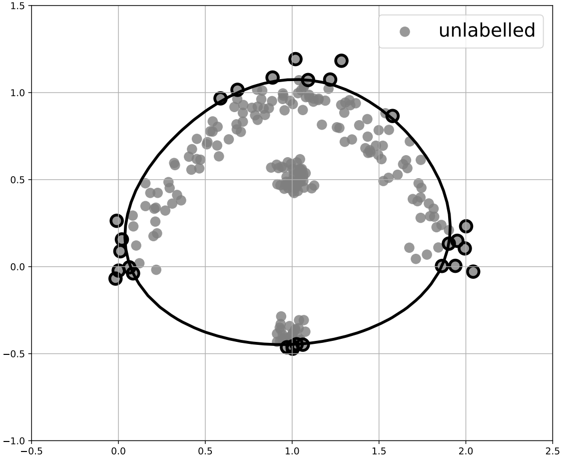
Introduction of a user feedback

- Semi-supervised context: unlabelled data $\mathcal{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$, labelled normal data $\mathcal{Y} = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_n\}$ and labelled anomalies $\mathcal{Z} = \{\boldsymbol{z}_1, ..., \boldsymbol{z}_n\}$ (e.g., resulting from user feedback)
- One-class SVM with user feedback

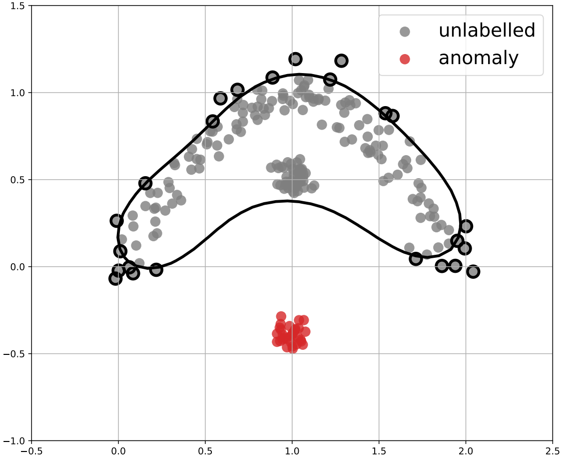
$$\arg\min_{\boldsymbol{w},\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C_1 \sum_{i=1}^{n_1} \xi_i + C_2 \sum_{l=1}^{n_2} \zeta_l + C_3 \sum_{p=1}^{n_3} \tau_p$$

$$\begin{aligned} \text{s.t. } & \boldsymbol{w}^T \Phi(\boldsymbol{x}_i) \geq 1 - \xi_i \text{ and } \xi_i \geq 0 \quad \text{unlabeled data} \\ & \boldsymbol{w}^T \Phi(\boldsymbol{y}_l) \geq 1 - \zeta_l \text{ and } \zeta_l \geq 0 \quad \text{labeled normal} \\ & \boldsymbol{w}^T \Phi(\boldsymbol{z}_p) \leq 1 + \tau_p \text{ and } \tau_p \geq 0 \quad \text{labeled anomalies} \end{aligned}$$

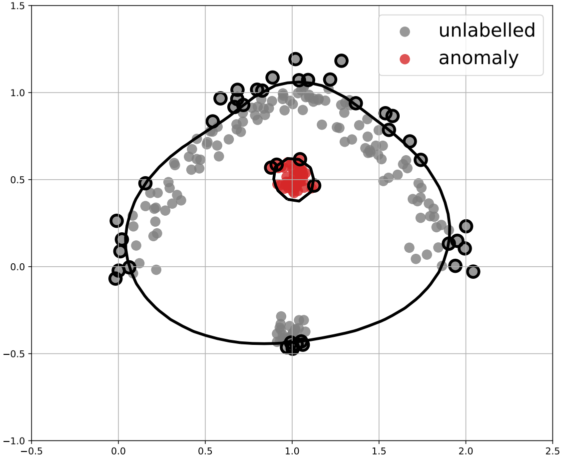
Examples



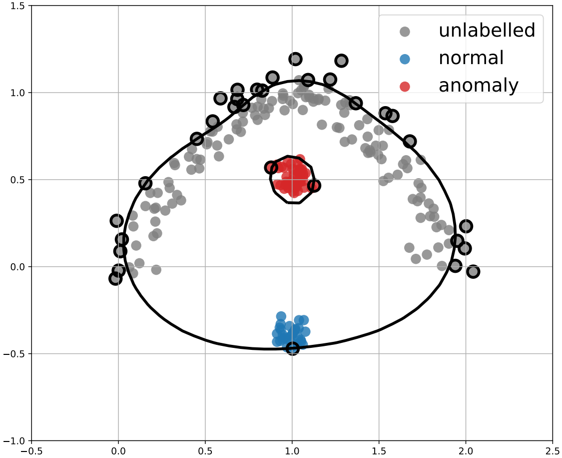
Examples



Examples



Examples



Support Vector Data Description (Tax and Duin, 1999)

Find a sphere of center $\boldsymbol{c}$ and radius R that encloses most of the data objects.

Optimization problem

$$\text{minimize } R^2 + C\sum_{i=1}^{n} \xi_i$$

$$\text{with the constraints } (\boldsymbol{x}_i - \boldsymbol{c})^T(\boldsymbol{x}_i - \boldsymbol{c}) \leq R^2 + \xi_i \; \xi_i \geq 0, \forall i$$

Optimization

Lagrangian

$$L\left(R, \boldsymbol{c}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}\right) = R^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i\left[R^2 + \xi_i - (\boldsymbol{x}_i - \boldsymbol{c})^T(\boldsymbol{x}_i - \boldsymbol{c})\right] - \sum_{i=1}^{n}\beta_i\xi_i$$

Set to zero the partial derivatives of L with respect to the primal variables $\boldsymbol{c}$, R and $\boldsymbol{\xi}$ yields

$$\boldsymbol{c} = \sum_{i=1}^{n}\alpha_i\boldsymbol{x}_i, \sum_{i=1}^{n}\alpha_i = 1 \text{ and } \alpha_i = C - \beta_i \leq C, \forall i$$

Dual problem

After replacing the expression of $\boldsymbol{c}$ in the Lagrangian, we obtain

$$U\left(\boldsymbol{\alpha}\right) = \sum_{i=1}^{n}\alpha_i\boldsymbol{x}_i^T\boldsymbol{x}_i - \sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j\boldsymbol{x}_i^T\boldsymbol{x}_j$$

that has to be maximized in the domain defined by $\sum_{i=1}^{n}\alpha_i = 1$ and $0 \leq \alpha_i \leq C$.

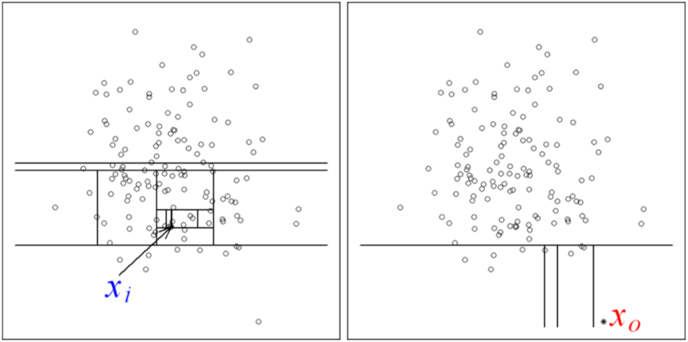
Summary

Anomaly detection

- Classes of anomalies
- Algorithms
  - Distance-based algorithms
    - LoOF and LOOP
    - Discords
  - Domain-based algorithms
    - One-Class SVM
  - Isolation Forest
  - Reconstruction-based algorithms
  - Online anomaly detection

Principle of isolation forests [Liu, 2008]

- Isolate each point by a random partitioning: an anomaly will be isolated faster than a nominal point



(a) Isolating x_i (b) Isolating x_o

How to built random trees?

Initial strategy proposed in the paper by Liu

For $\mathcal{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$ with $\boldsymbol{x}_i \in \mathbb{R}^d$, a sample of ψ instances $\mathcal{X}' \subset \mathcal{X}$ (ψ : subsample size) is used to build an isolation tree.

For each vector $\boldsymbol{x}_i \in \mathcal{X}'$

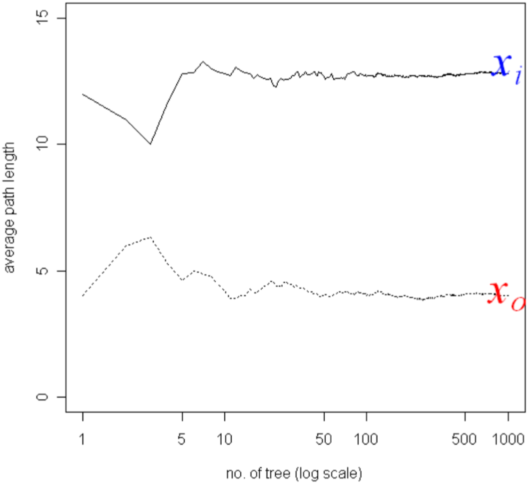
- Select one feature randomly F_k
- Compute the minimum and maximum of this feature denoted as $\max_k$ and $\min_k$
- Divide the space into two parts corresponding to $F_k < s_k$ and $F_k \geq s_k$, where s_k is uniformly distributed in $]\min_k, \max_k[$
- Repeat the process until $\boldsymbol{x}_i$ has been isolated

Average the numbers of steps obtained with different trees

$$E[h(\boldsymbol{x}_i)]$$

Note that it is NOT an expectation!!

Length of an average path



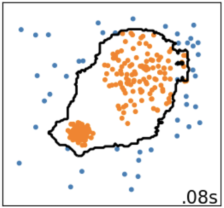
(c) Average path lengths converge

Anomaly score

- Definition

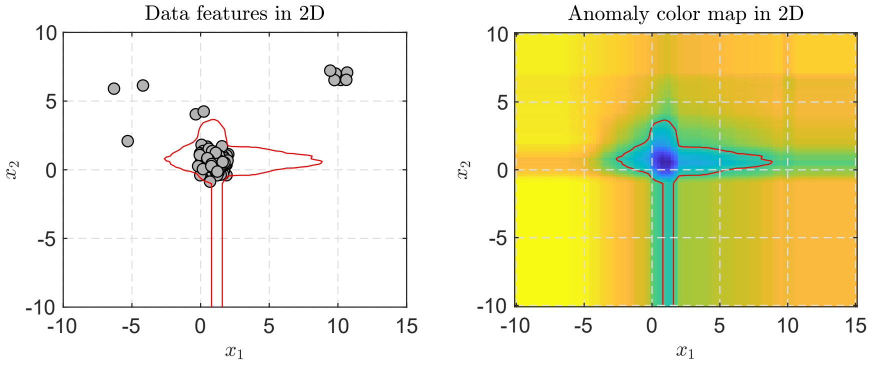
$$s(\boldsymbol{x}_i, \psi) = 2^{-\frac{E[h(\boldsymbol{x}_i)]}{c(\psi)}}$$

where $E[h(\boldsymbol{x}_i)]$ is the average path length for $\boldsymbol{x}_i$ and $c(\psi)$ is the average length of a path for a tree with ψ instances ($c(\psi)$ available in [Liu, 2008])
  - if $E[h(\boldsymbol{x}_i)] = c(\psi)$ then $s(\boldsymbol{x}_i, \psi) = 0.5$ (uncertainty)
  - if $E[h(\boldsymbol{x}_i)]$ tends to 0, then $s(\boldsymbol{x}_i, \psi)$ tends to 1 ($\boldsymbol{x}_i$ is an anomaly)
  - if $E[h(\boldsymbol{x}_i)]$ tends to $\psi - 1$, then $s(\boldsymbol{x}_i, \psi)$ tends to 0 ($\boldsymbol{x}_i$ is normal)
- Separating curve: defined using the averaged lengths of the paths

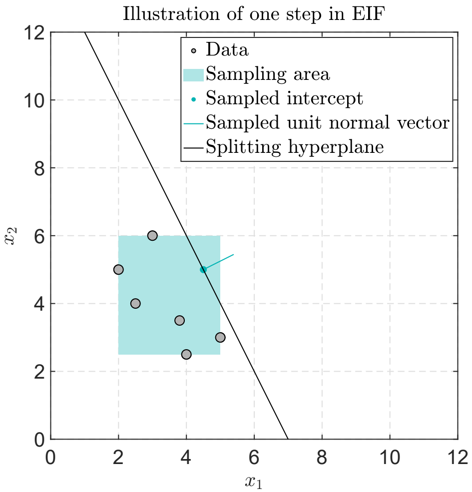


Orange samples: $s(\boldsymbol{x}_i, \psi) \leq 0.5$, blue samples: $s(\boldsymbol{x}_i, \psi) > 0.5$.

Problem with isolation forest



Extended Isolation Forest



Generalized Isolation Forest

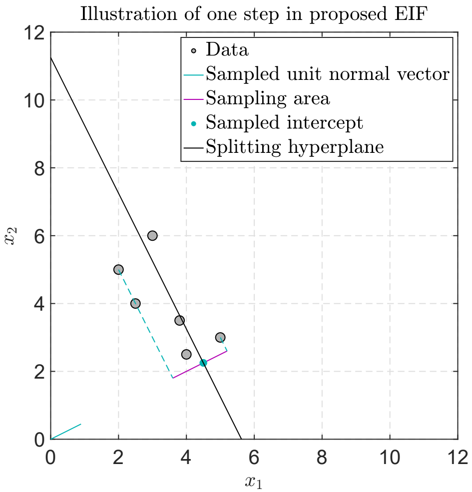
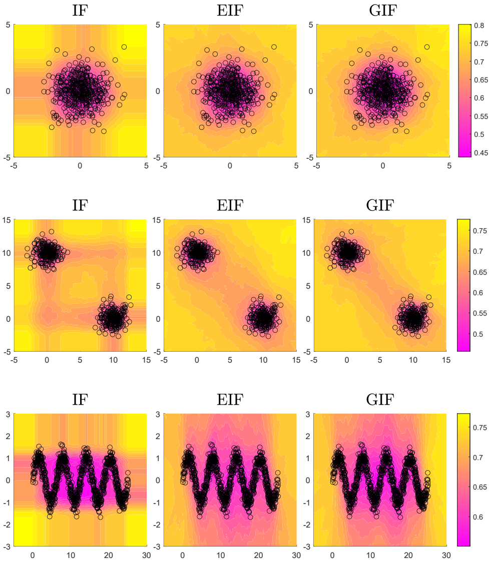


Illustration on Synthetic Satasets



Computation times in seconds

Dataset	EIF	GIF
Pen Local	2.081 ± 0.0998	$\mathbf{1.11} \pm 0.0731$
Forest Cover	1.66 ± 0.0692	$\mathbf{0.981} \pm 0.0624$
Speech	10.376 ± 0.839	$\mathbf{4.729} \pm 0.472$
Shuttle	1.2 ± 0.0615	$\mathbf{0.856} \pm 0.0381$
Mammography	1.113 ± 0.0805	$\mathbf{0.776} \pm 0.0578$
Breast Cancer	1.349 ± 0.0514	$\mathbf{0.857} \pm 0.0454$
Aloi	0.916 ± 0.0548	$\mathbf{0.699} \pm 0.0505$
ANN Thyroid	1.103 ± 0.0525	$\mathbf{0.778} \pm 0.0463$
Letter	2.027 ± 0.1005	$\mathbf{1.112} \pm 0.0657$
Cardio	1.378 ± 0.0639	$\mathbf{0.912} \pm 0.0605$
Pen Global	2.039 ± 0.0983	$\mathbf{1.079} \pm 0.0654$
Satellite	1.963 ± 0.0811	$\mathbf{1.145} \pm 0.058$
Ionosphere	2.009 ± 0.074	$\mathbf{0.875} \pm 0.0581$

Summary

Anomaly detection

- Classes of anomalies
- Algorithms
  - Distance-based algorithms
    - LoOF and LOOP
    - Discords
  - Domain-based algorithms
    - One-Class SVM
  - Isolation Forest
  - Reconstruction-based algorithms
    - Subspace-based methods
    - Neural network-based approaches
  - Online anomaly detection

Outlier detection using PCA [Shyu, 2003]

- Robust estimation of the mean and correlation matrix of normal data
  - Conventional estimators of the mean and correlation matrix: $\bar{\boldsymbol{x}}$ and $\boldsymbol{\Sigma}$
  - Remove the vectors with the γth largest values of

$$d_i^2 = (\boldsymbol{x}_i - \bar{\boldsymbol{x}})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \bar{\boldsymbol{x}})$$

    These vectors are more likely to be anomalies!
  - Recompute the mean and the correlation matrix $\boldsymbol{\Sigma}$ of the remaining vectors.
- Principal component analysis (PCA) of the vectors $\boldsymbol{x}_i$
- Compute two test statistics from the projected vector $\boldsymbol{y}_i = \boldsymbol{P}\boldsymbol{x}_i$

$$T_{i,q} = \sum_{j=1}^{q} \frac{y_{ij}^2}{\lambda_j} \quad U_{i,p} = \sum_{j=p-r+1}^{p} \frac{y_{ij}^2}{\lambda_j}$$

  where $\lambda_1, ..., \lambda_q$ are the q largest singular values of $\boldsymbol{\Sigma}$ (q such that 50% of the inertia is preserved), and $\lambda_{p-r+1}, ..., \lambda_p$ are the r smallest values of $\boldsymbol{\Sigma}$. Note that $T_{i,q}$ estimates the distance between $\boldsymbol{x}_i$ and the mean vector whereas $U_{i,p}$ identifies vectors that have correlation structures different from the normal data.
- Declare that $\boldsymbol{x}_i$ is an anomaly if $T_{i,q} > c_1$ or if $U_{i,q} > c_2$

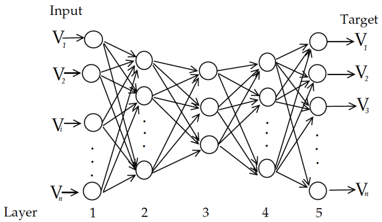
Summary

Anomaly detection

- Classes of anomalies
- Algorithms
  - Distance-based algorithms
    - LoOF and LOOP
  - Domain-based algorithms
    - One-Class SVM
  - Isolation Forest
  - Reconstruction-based algorithms
    - Subspace-based methods
    - Neural network-based approaches
  - Online anomaly detection

Outlier detection using RNNs [Hawkins, 2002]

- Architecture of replicator neural networks



- tanh activation functions for layers 2 and 4
- staircase activation function for layer 3
- linear or sigmoidal activation function for the output layer

How to use RNNs for outlier detection?

- Weights
  The weights of the hidden layers are optimized to minimize the reconstruction error across all training patterns.

$$\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}(x_{ij}-o_{ij})^2$$

  where m is the number of vectors in the database, n is the number of features of $\boldsymbol{x}_i$, x_{ij} and o_{ij} are the jth features of the ith data record $\boldsymbol{x}_i$ at the input and output of the network.

- Outlier factor for the ith data record

$$\mathsf{OF}_i = \frac{1}{n}\sum_{j=1}^{n}(x_{ij}-o_{ij})^2.$$

  The anomalies are the samples that are not well reconstructed by the network!

Summary

Anomaly detection

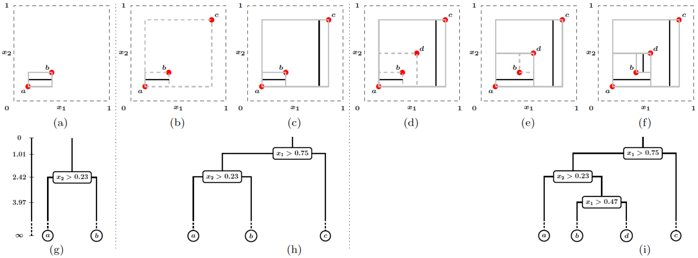
- Classes of anomalies
- Algorithms
  - Distance-based algorithms
    - LoOF and LOOP
  - Domain-based algorithms
    - One-Class SVM
  - Isolation Forest
  - Reconstruction-based algorithms
    - Neural network-based approaches
    - Subspace-based methods
  - Online anomaly detection

Online anomaly detection

- One-class SVM
  Exploit the structure of the one-class SVM problem to find a subspace minimizer for an $(n+1)$-point SVM problem by using the solution of the n-point problem. This can be done using active-set quadratic programming (Gao, 2015) or incremental/decremental learning (Diehl, 2003)
- Online decision trees
  - Random Forest (Saffari, 2009): Duplicate a new observation (number of replications distributed according to a Poisson $\mathcal{P}(1)$ distribution) and classify these observations using the existing tree. A node is divided into two branches if 1) there is a minimum number of observations in this node, 2) the Gini index is sufficiently reduced after separation. A node is suppressed when its out-of-bag error is too large.

Online anomaly detection

- One-class SVM
- Online decision trees
  - Random Forest (Saffari, 2009)
  - Mondrian Forests (Lakshminarayanan, 2014): Divide the observation space into hypercubes as a Mondrian painting and update this decision tree when a new observation is arriving by continuing an existing split or by creating new branches inside an existing split.



References on anomaly detection

Surveys

- V. Chandola and A. Banerjee and V. Kumar, Anomaly detection: a survey, ACM Computing Surveys, vol. 41, no. 3, pp. 1-62, 2009.
- M. A. F. Pimentel, D. A. Clifton and L. Tarassenko, A review of novelty detection, Signal Processing, vol. 99, pp. 215-249, 2014.

LOF, LoOP and Discords

- M. M. Breunig, H.-P. Kriegel, R. T. Ng and J. Sander, LOF: Identifying Density-Based Local Outliers, Proc. Int. Conf. Management of Data (SIGMOD), Dallas, TX, USA, 2000.
- H. P. Kriegel, P. Kröger, E. Schubert, and A. Zime, LoOP: Local outlier probabilities, Proc. Conf. Information Knowledge Management (CIKM), Hong-Kong, China, 2009.
- E. Keogh, J. Lin and A. Fu, HOT SAX: Finding the Most Unusual Time Series Subsequence: Algorithms and Application, Proc. Int. Conf. Data Mining (ICDM), Houston, Texas, Nov. 27-30, 2005.

References on anomaly detection

One-Class SVM

- B. Schölkopf, J. Platt, J. Shawe-Taylor, A. Smola, and R. Williamson, Estimating the Support of a High-Dimensional Distribution, Neural Computation, vol. 13, no. 7, pp. 1443-1471, 2001.
- D. Tax and R. Duin, Support Vector Domain Description, Pattern Recognition Letters, vol. 20, pp. 1191-1199, 1999.

Isolation Forests, extended and generalized isolation forests

- F. T. Liu, K. M. Ting and Z.-H. Zhou, Isolation Forest, Proc. IEEE Int. Conf. Data Mining, Pisa, Italy, 2008.
- S. Hairi, M. C. Kind and R. J. Brunner, Extended Isolation Forest, IEEE Trans. Knowl. Data Eng., vol. 33, no. 4, April 2021.
- J. Lesouple, C. Baudoin, M. Spigai and J.-Y. Tourneret, Generalized Isolation Forest for Anomaly Detection, Pattern Recognition Letters, vol. 149, pp. 109-119, 2021.

References on anomaly detection

Reconstruction algorithms

- M.-L. Shyu, S.-C. Chen, K. Sarinnapakorn and L. Chang, A Novel Anomaly Detection Scheme Based on Principal Component Classifier, Proc. Int. Conf on Data Mining, Melbourne, Florida, USA, Nov. 19-22, 2003.
- S. Hawkins, H. He, G. Williams and R. Baxter, Outlier Detection Using Replicator Neural Networks, Data Warehouse Knowledge Discovery, vol. 2454, pp. 170-180, 2002.

Online one-class SVM

- C. P. Diehl and G. Cauwenberghs, SVM Incremental Learning, Adaptation and Optimization, Proc. Int. Joint Conf. Neural Networks (IJCNN), Portland, OR, USA, July 20-24, 2003.
- K. Gao, Online One-class SVMs with Active-set Optimization for Data Streams, Proc. Int. Conf. Machine Learning and Applications (ICMLA), Miami, FL, USA, Dec. 9-11, 2015

References on anomaly detection

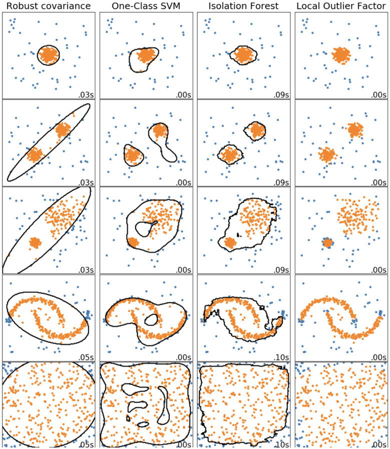
Online random forests

- A. Saffari et al., On-line Random Forests, Proc. Int. Conf. Computer Vision (ICCV), Kyoto, Japan, Sep. 27-Oct. 04, 2009.
- B. Lakshminarayanan, D. M. Roy and Y. W. Teh, Mondrian Forests: Efficient Online Random forests, Proc. Advances in Neural Information Processing Systems (NIPS), Montreal, Canada, Dec. 8-13, 2014.

Thanks for your attention!

Anomaly detection

Scikit learn examples



Lien: https://scikit-learn.org/0.21/auto_examples/plot_anomaly_comparison.html