Functional Isolation Forest

Pavlo Mozharovskyi

joint with

Guillaume Staerman, Stephan Clémençon, and Florence D'Alché-Buc

LTCI, Telecom Paris, Institut Polytechnique de Paris

Séminaire de statistique appliquée du CNAM

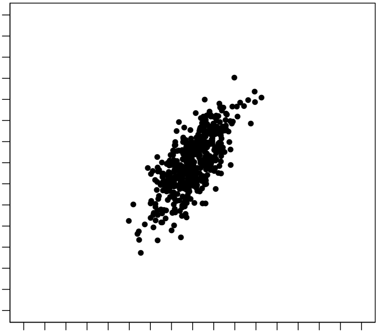
Paris, July 24, 2022

Contents

Contents

A real task

Regard two measurements during a test in a production process:

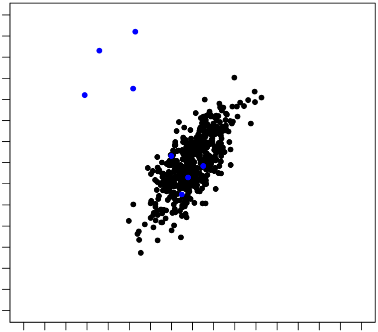


Given **training data**, polluted or not with anomalies:

- detect **anomalies** in the given data.

A real task

Regard two measurements during a test in a production process:



Given **training data**, polluted or not with anomalies:

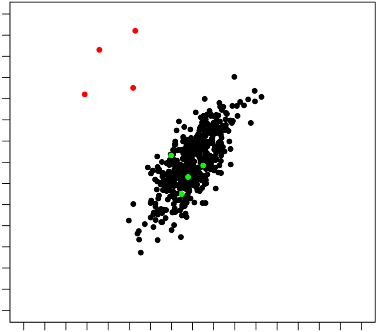
- detect **anomalies** in the given data.

For **new data**, determine:

- Whether new observations are **normal** data or **anomalies**?

A real task

Regard two measurements during a test in a production process:



Given **training data**, polluted or not with anomalies:

- detect **anomalies** in the given data.

For **new data**, determine:

- Whether new observations are normal data or anomalies?

Multivariate framework

- A training data set:

$$\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\} \subset \mathbb{R}^d$$

of observations in the d-dimensional Euclidean space.

Multivariate framework

- A training data set:

$$\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\} \subset \mathbb{R}^d$$

  of observations in the d-dimensional Euclidean space.

- Typical example: a table from a data base, with lines being observations (=individuals, items,...).

Multivariate framework

- A training data set:

$$\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\} \subset \mathbb{R}^d$$

  of observations in the d-dimensional Euclidean space.

- Typical example: a table from a data base, with lines being observations (=individuals, items,...).

- Construct a decision function:

$$\mathbb{R}^d \to \{-1, +1\} : \boldsymbol{x} \mapsto g(\boldsymbol{x}),$$

  which attributes to any (possible) $\boldsymbol{x} \in \mathbb{R}^d$ a label whether it is an anomaly (*e.g.*, $+1$) or a normal observation (*e.g.*, -1).

Multivariate framework

- A training data set:

$$\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\} \subset \mathbb{R}^d$$

  of observations in the d-dimensional Euclidean space.

- Typical example: a table from a data base, with lines being observations (=individuals, items,...).

- Construct a decision function:

$$\mathbb{R}^d \to \{-1, +1\} : \boldsymbol{x} \mapsto g(\boldsymbol{x}),$$

  which attributes to any (possible) $\boldsymbol{x} \in \mathbb{R}^d$ a label whether it is an anomaly (*e.g.*, $+1$) or a normal observation (*e.g.*, -1).

- It is more useful to provide an ordering on $\mathbb{R}^d$:

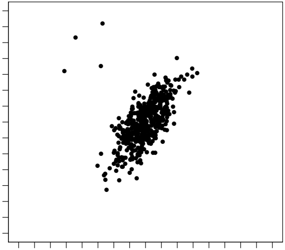
$$\mathbb{R}^d \to \mathbb{R} : \boldsymbol{x} \mapsto g(\boldsymbol{x}),$$

  such that abnormal observations obtain higher anomaly score.

Anomaly Detection (AD)

- What is Anomaly detection?
  - Identify unusual patterns that do not conform to expected behavior.

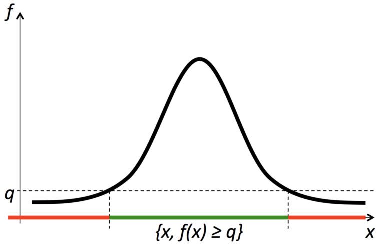
- Applications : Network intrusions, credit card fraud detection, insurance, finance, military surveillance, predictive maintenance, medical monitoring.



Unsupervised Anomaly Detection

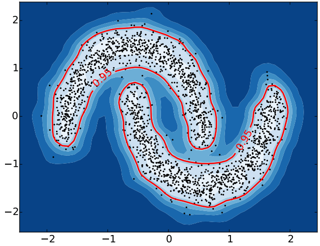
Unsupervised Anomaly Detection: data are unlabelled. We suppose that all data come from the same distribution μ and that anomalies are very rare, *i.e.*, belongs to the low density regions.

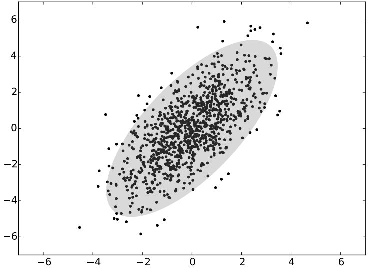
If μ admits a density f w.r.t. a measure ρ, the goal of anomaly detection can be formulated as the recovery of upper-level sets $\{x : f(x) \geq q\}$, $q \geq 0$.



Statistical inference

- Plug-in method : Estimation by $\{x : \hat{f}(x) \geq q\}$.
- Direct methods : Build a score function $s : \mathcal{X} \to \mathbb{R}$ such that $\{x : s(x) \geq q\}$ is close to $\{x : f(x) \geq q\}$.





Unsupervised AD in practice

- How to do it in practice?

Step 1. Learn a score function $s : \mathcal{X} \to \mathbb{R}$ which assigns a score to each data.

Step 2. Find the best treshold to construct a decision function which separates "normal" and "abnormal" data and then induces two regions.

Step 3. **Detect anomalies** among new observations.

Contents

Isolation forest (Liu, Ting, Zhou; 2008)

- **Isolation forest** (Liu, Ting, Zhou; 2008) is an anomaly detection method inherited from the famous random forest algorithm (Breiman, 2001).
- Since no supervised feedback is given, isolation forest is based on purely random (uniform) variable-based partitioning.

Isolation forest (Liu, Ting, Zhou; 2008)

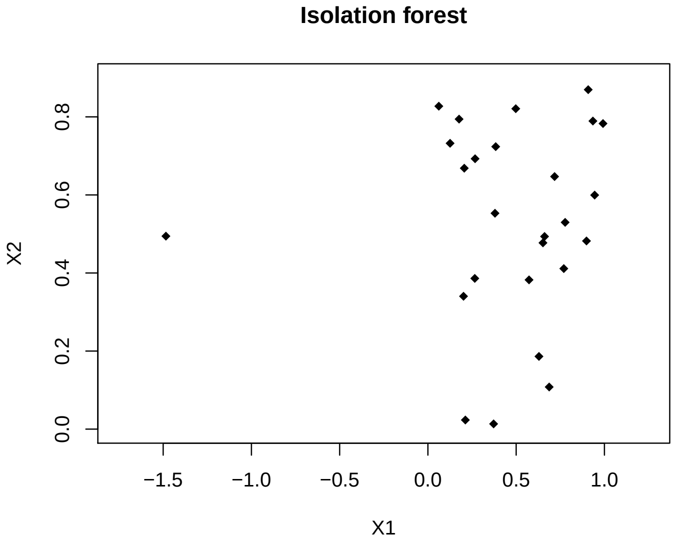
- **Isolation forest** (Liu, Ting, Zhou; 2008) is an anomaly detection method inherited from the famous random forest algorithm (Breiman, 2001).
- Since no supervised feedback is given, isolation forest is based on purely random (uniform) variable-based partitioning.
- **Main idea:** Outlying observations are isolated faster.

Isolation forest (Liu, Ting, Zhou; 2008)

- **Isolation forest** (Liu, Ting, Zhou; 2008) is an anomaly detection method inherited from the famous random forest algorithm (Breiman, 2001).
- Since no supervised feedback is given, isolation forest is based on purely random (uniform) variable-based partitioning.
- **Main idea:** Outlying observations are isolated faster.
- Tree-kind partitioning is done until "full isolation": outlying observations will have smaller depth (on an average) in the isolation tree.
- A monotone transform is usually applied to the aggregated estimate.
- To reduce both masking effect and computation cost, small-size sub-sampling is used instead of bootstrap.

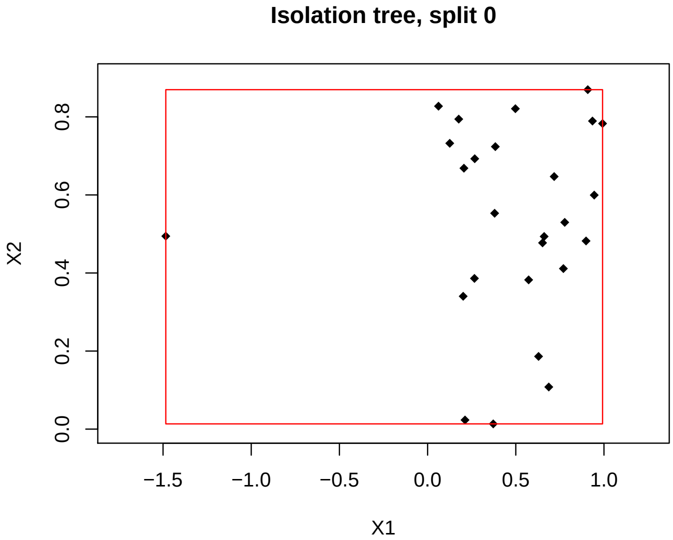
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



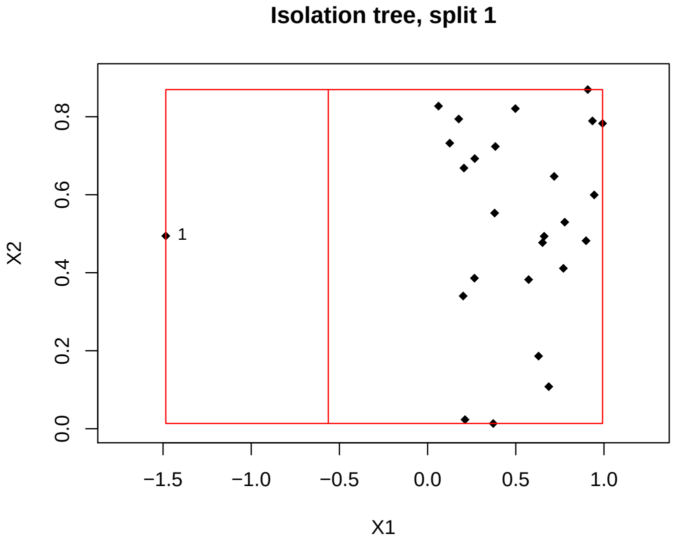
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



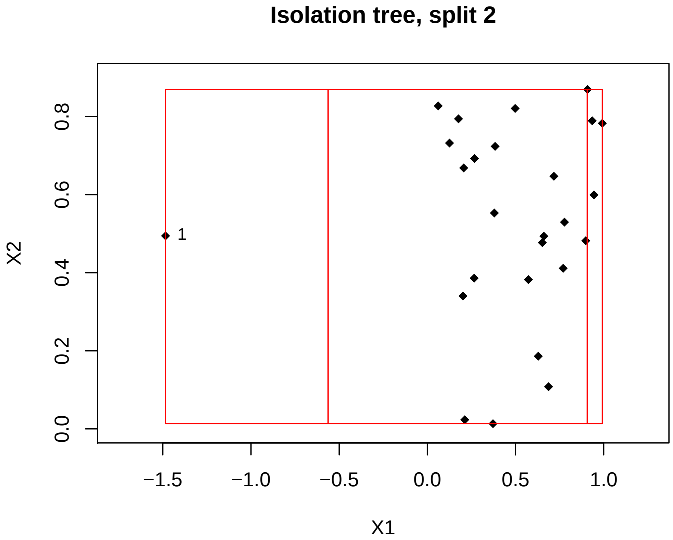
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



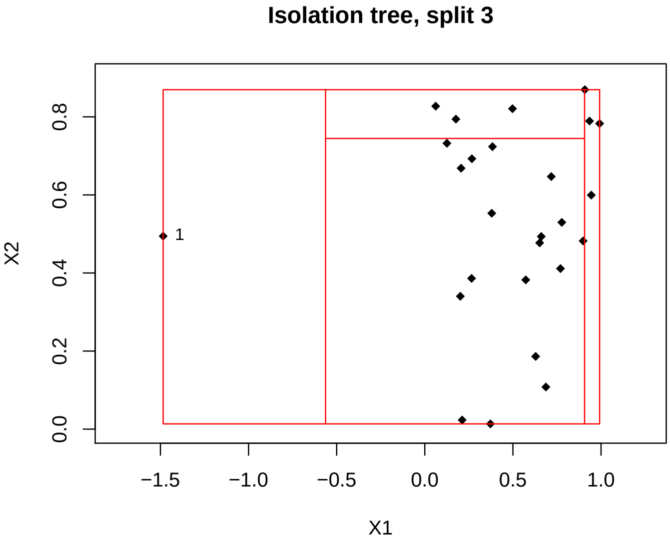
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



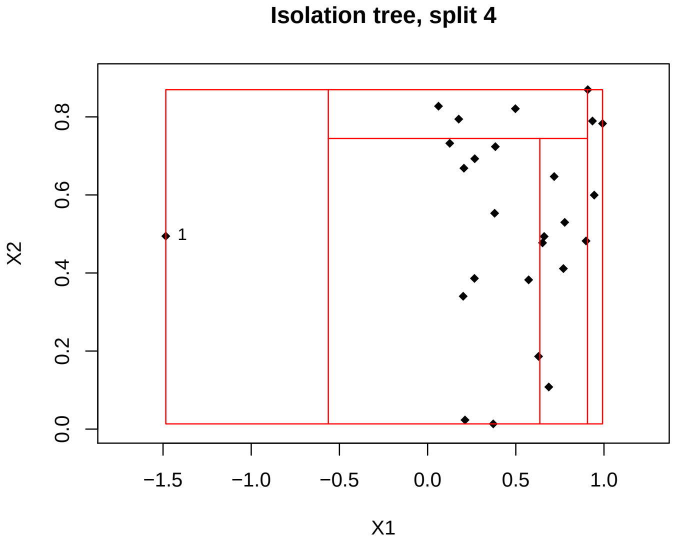
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



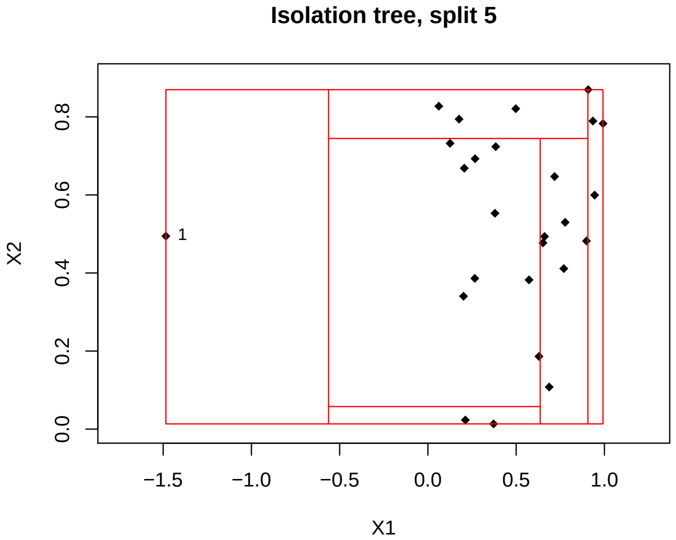
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



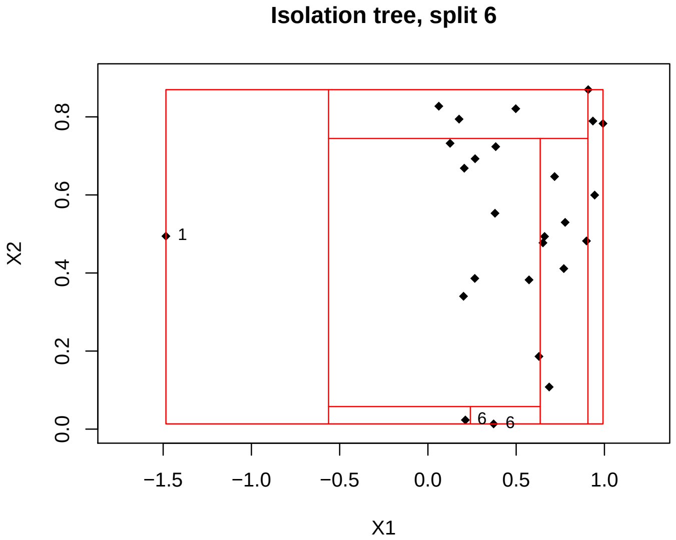
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



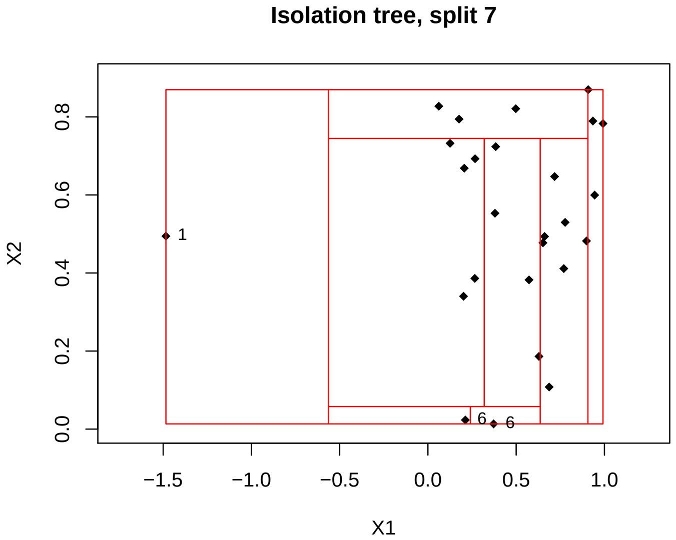
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



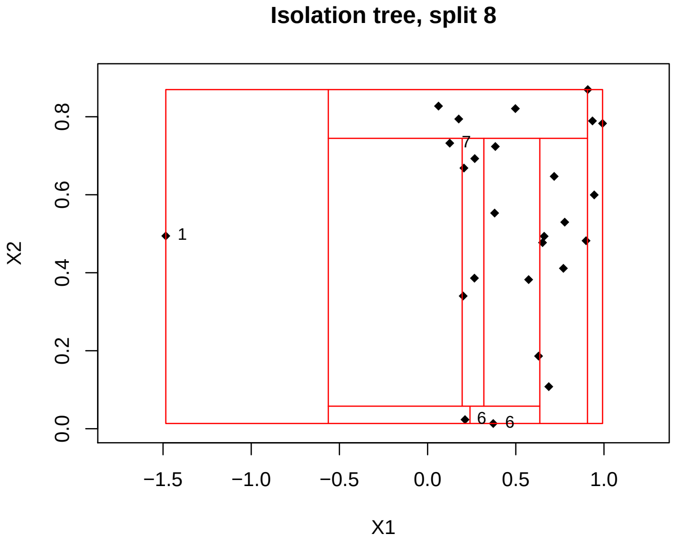
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



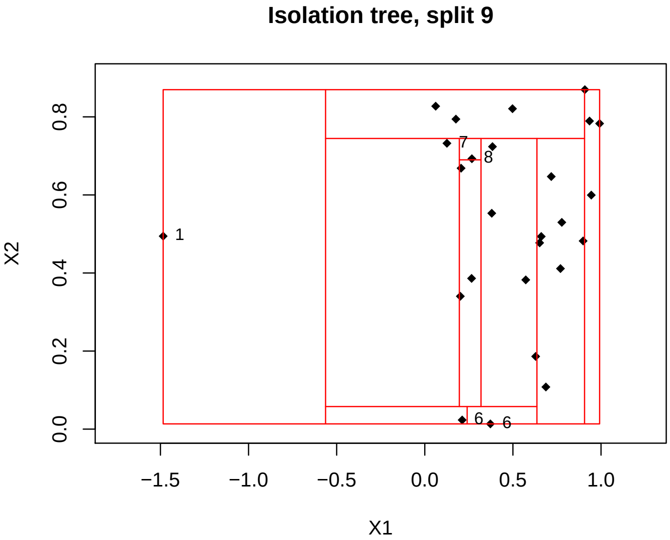
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



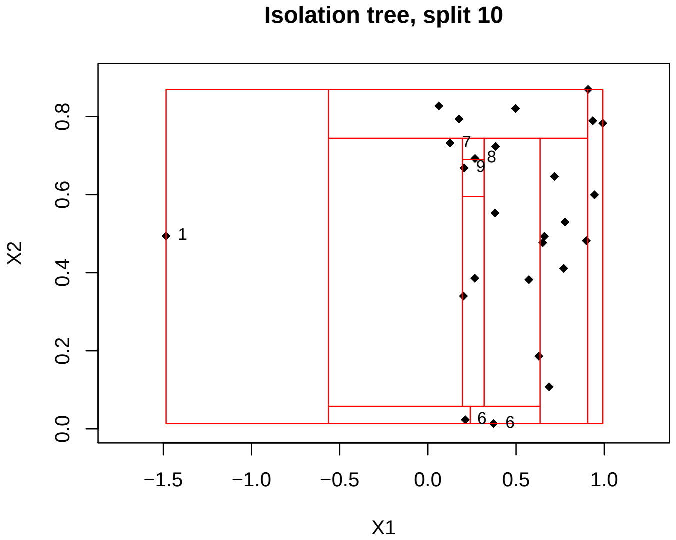
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



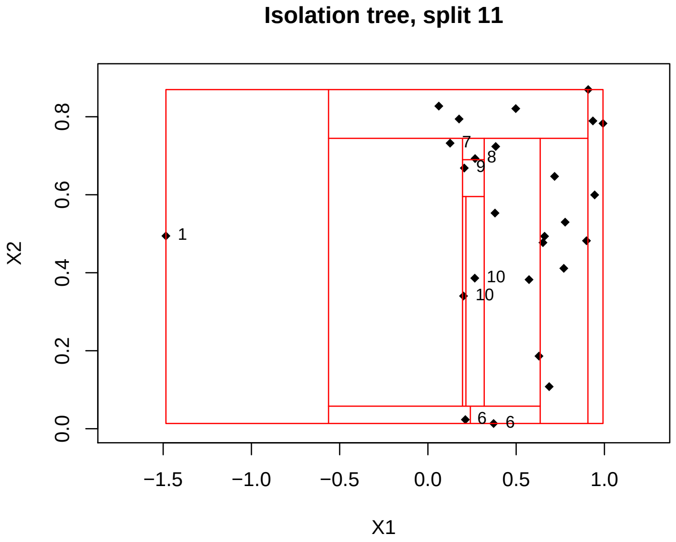
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



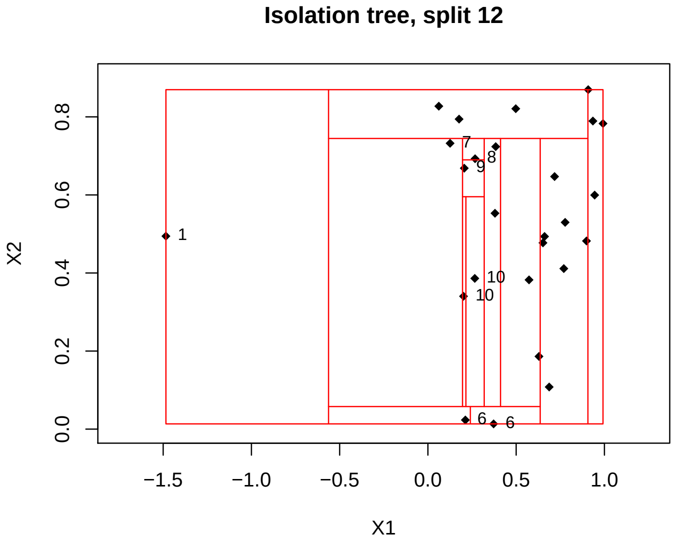
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



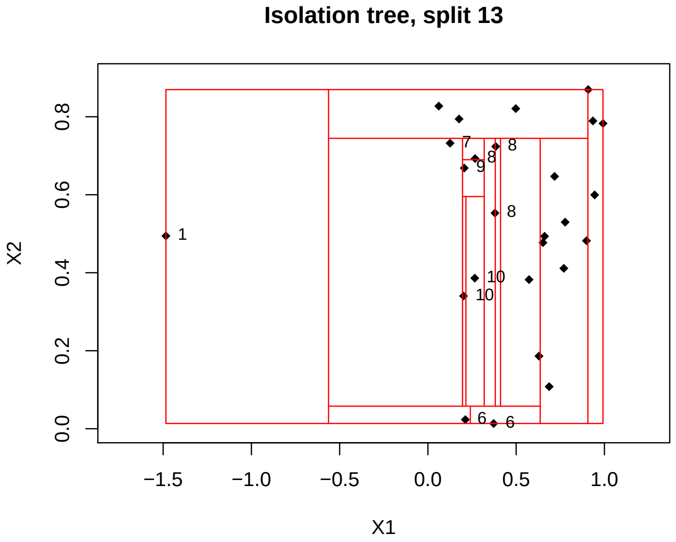
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



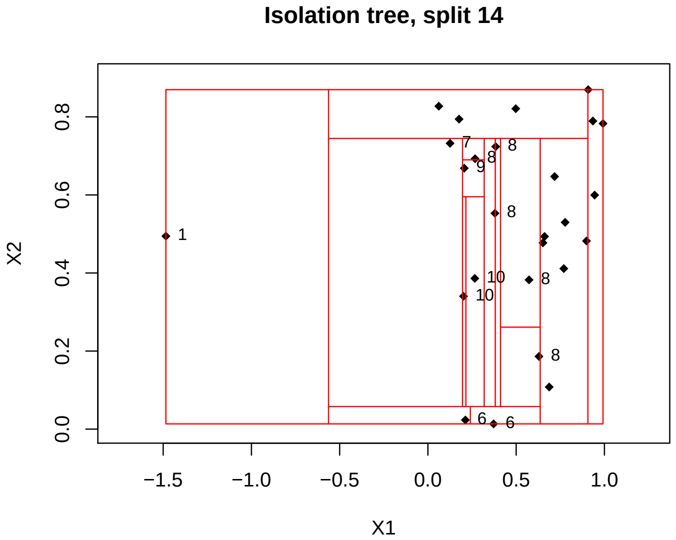
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



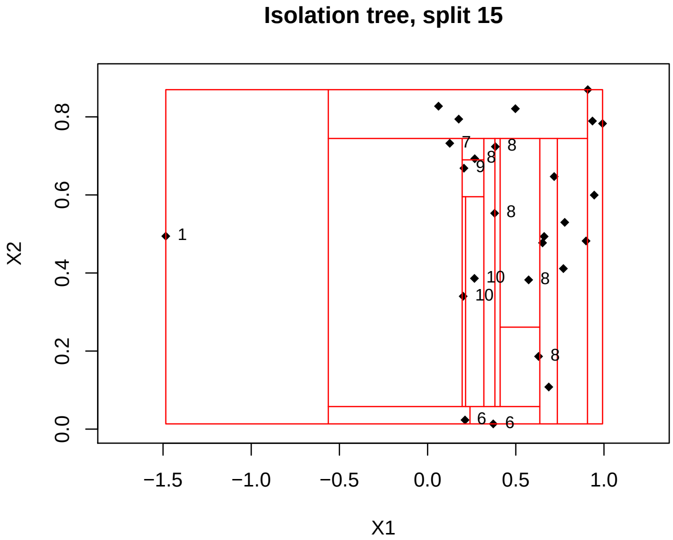
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



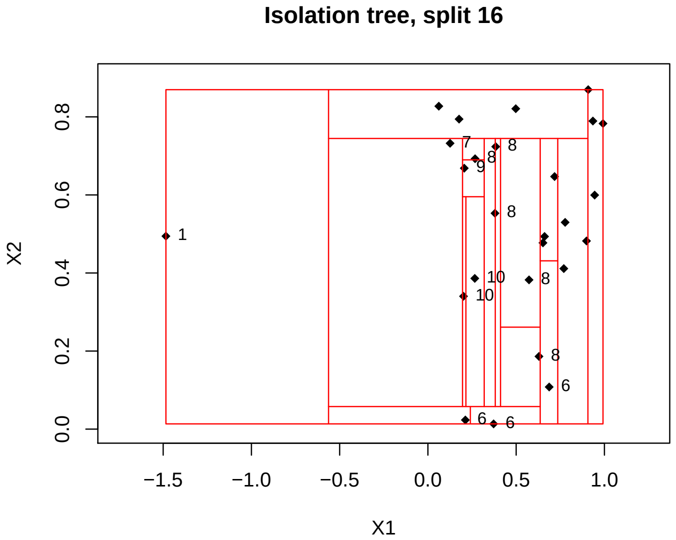
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



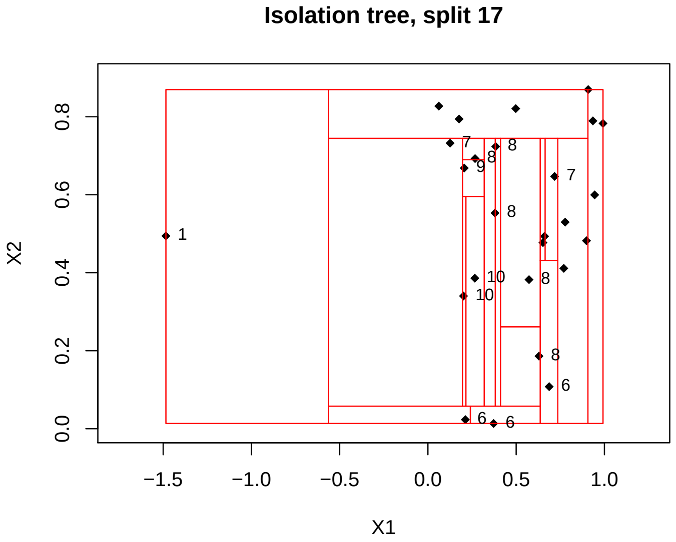
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



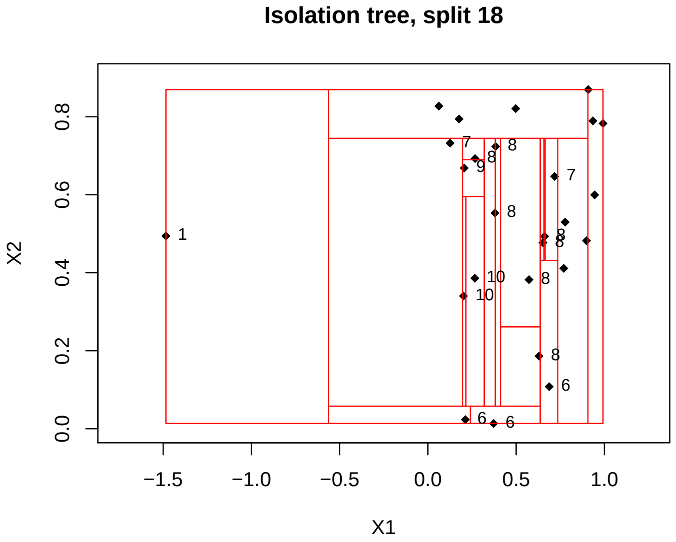
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



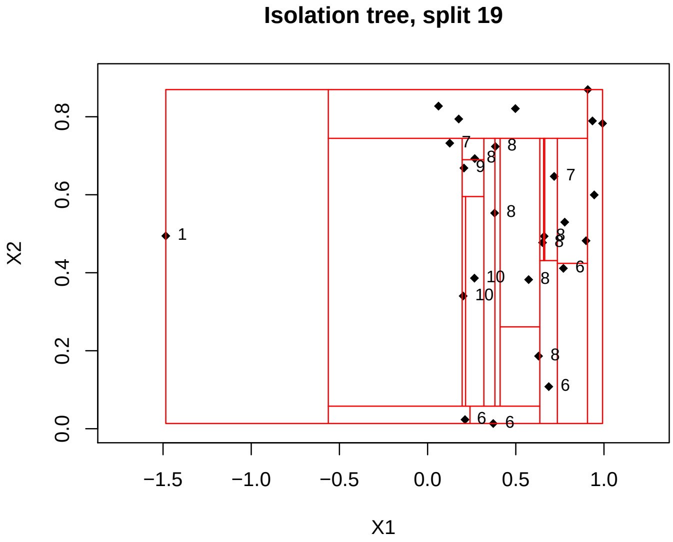
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



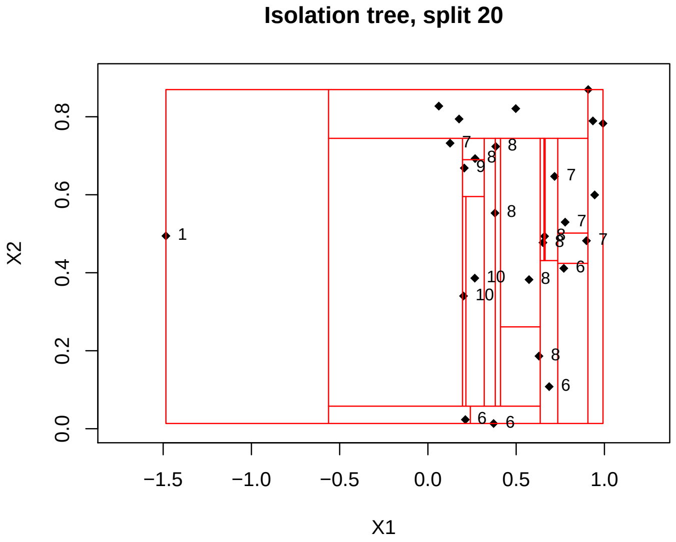
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



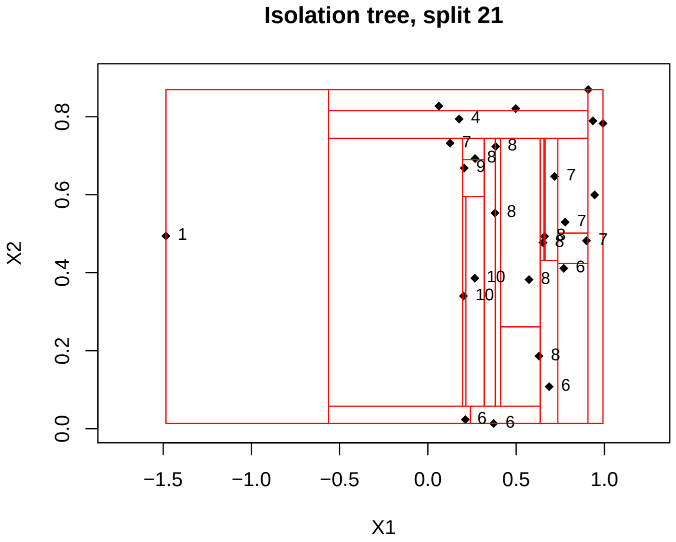
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



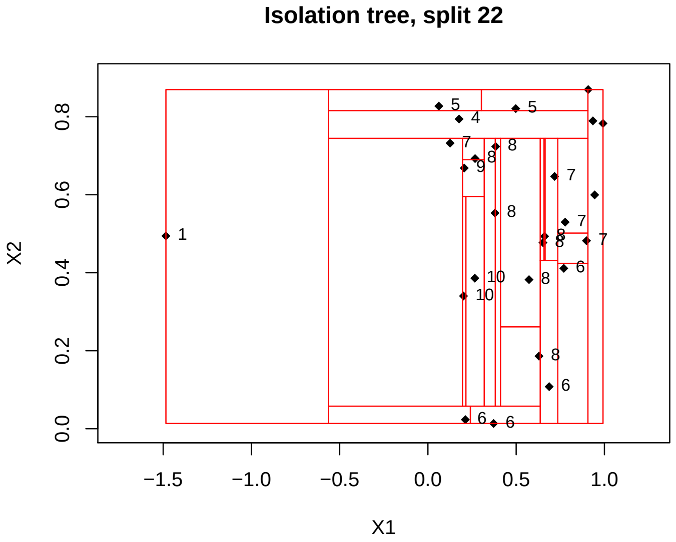
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



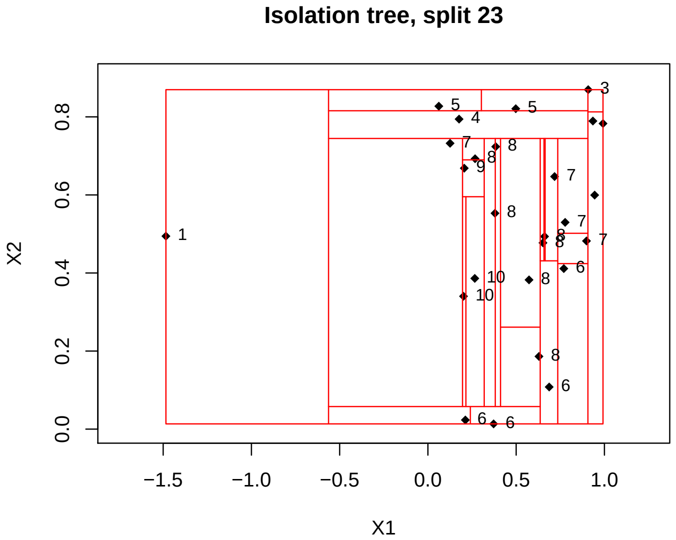
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



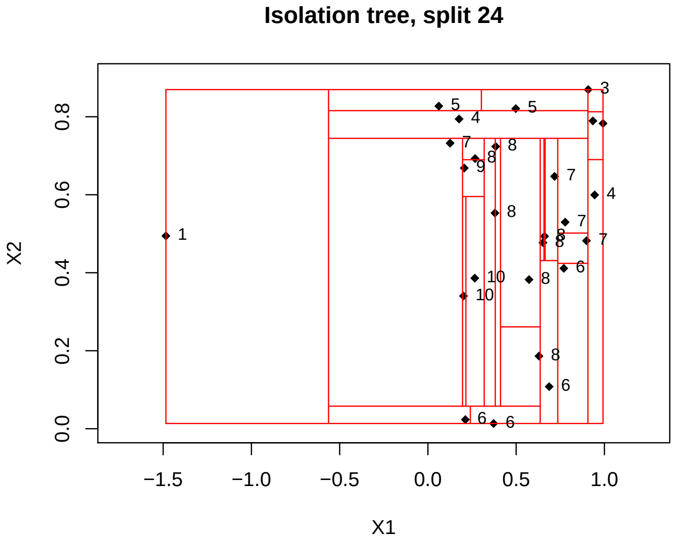
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



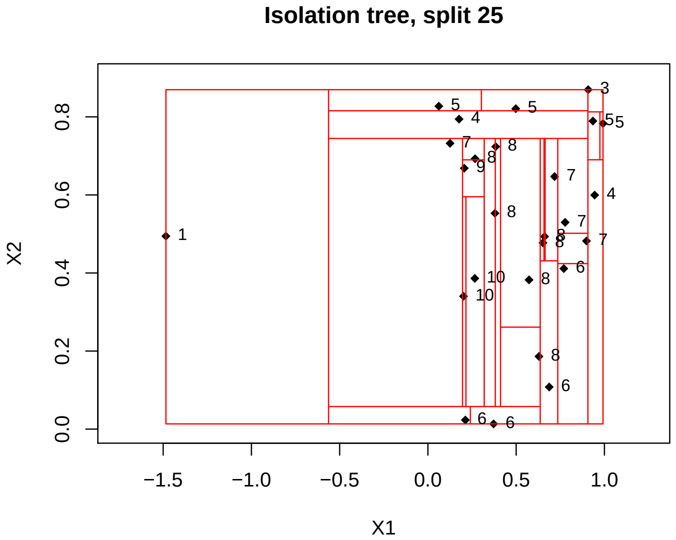
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



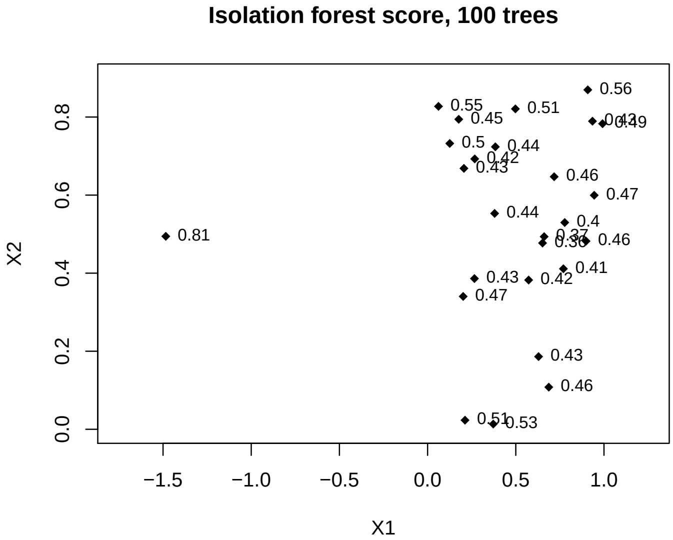
Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Isolation tree



Isolation forest (Liu, Ting, Zhou; 2008)

Illustration: Anomaly score



Contents

Functional Data Framework

- Let X be a functional random variable that takes its values in a functional space and $X_1, \ldots, X_n$ be an i.i.d. sample from X:

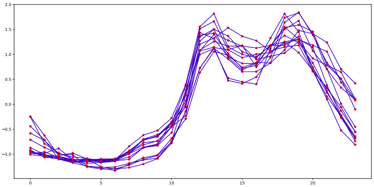
$$\begin{array}{rcccc} X & : & (\Omega,\ \mathcal{A},\ \mathbb{P}) & \longrightarrow & \mathcal{H}([0,1]) \\ & & \omega & \longmapsto & X(\omega) = (X_t(\omega))_{t\in[0,1]}. \end{array}$$

Functional Data Framework

- Let X be a functional random variable that takes its values in a functional space and $X_1, \ldots, X_n$ be an i.i.d. sample from X:

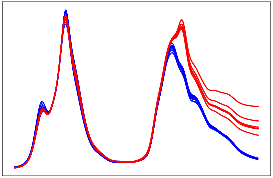
$$\begin{array}{cccccc} X & : & (\Omega,\ \mathcal{A},\ \mathbb{P}) & \longrightarrow & \mathcal{H}([0,1]) \\ & & \omega & \longmapsto & X(\omega) = (X_t(\omega))_{t\in[0,1]}. \end{array}$$

- Data are basically parametric curves, *i.e.*, data collected in quasi-real time.
- In practice, we only have access to a finite number of arguments/times, $\mathcal{S}_n = \{X_i(t_1), \ldots, X_i(t_p),\ \ 1 \leq i \leq n\}$ such that $0 \leq t_1 < \cdots < t_p \leq 1$.
- First step: reconstruct a functional object from time-series either by interpolation or basis decomposition.

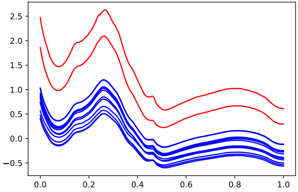


Anomaly Detection and functional data (Hubert et al,. 2015)

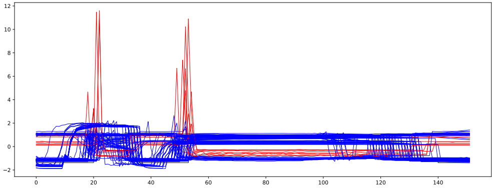
Shape anomalies



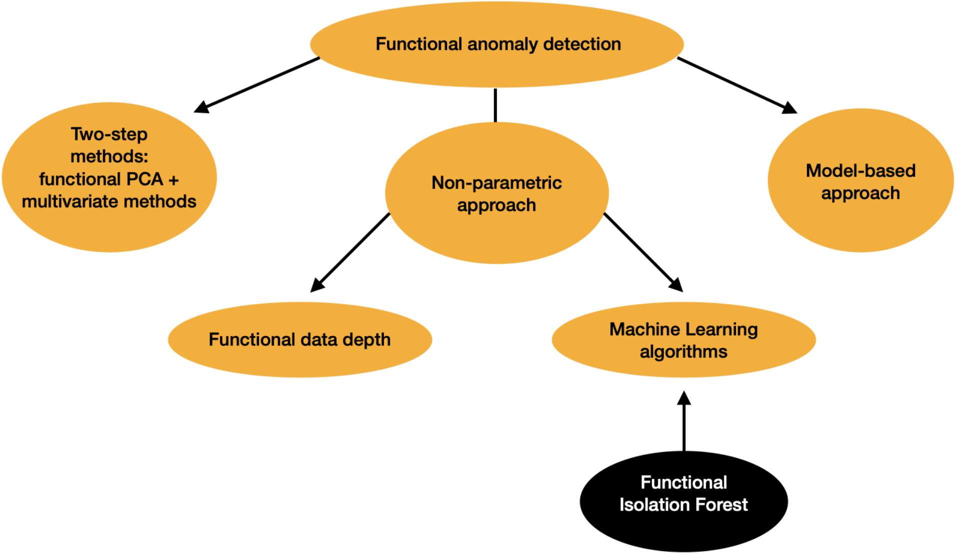
Shift anomalies



Isolated anomalies



Context of FAD contributions



Contents

Contents

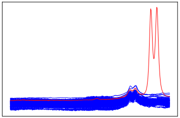
Functional Isolation Forest

- $X_1, \ldots, X_n$ are random variables in Hilbert space $\mathcal{H}$ and $\mathcal{D} \subset \mathcal{H}$.
- This ensemble learning algorithm builds a collection of binary tree based on a recursive and randomized tree-structured partitioning procedure.

Functional Isolation Forest

- $X_1, \ldots, X_n$ are random variables in Hilbert space $\mathcal{H}$ and $\mathcal{D} \subset \mathcal{H}$.
- This ensemble learning algorithm builds a collection of binary tree based on a recursive and randomized tree-structured partitioning procedure.

Step 1:



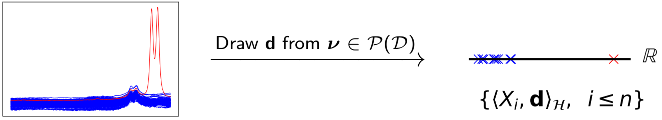
Draw $\mathbf{d}$ from $\boldsymbol{\nu} \in \mathcal{P}(\mathcal{D})$ $\longrightarrow$

$\{\langle X_i, \mathbf{d} \rangle_{\mathcal{H}}, \;\; i \leq n\}$

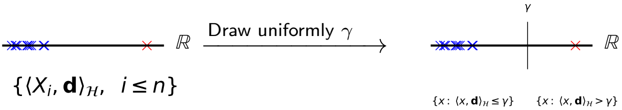
Functional Isolation Forest

- $X_1, \dots, X_n$ are random variables in Hilbert space $\mathcal{H}$ and $\mathcal{D} \subset \mathcal{H}$.
- This ensemble learning algorithm builds a collection of binary tree based on a recursive and randomized tree-structured partitioning procedure.

Step 1:



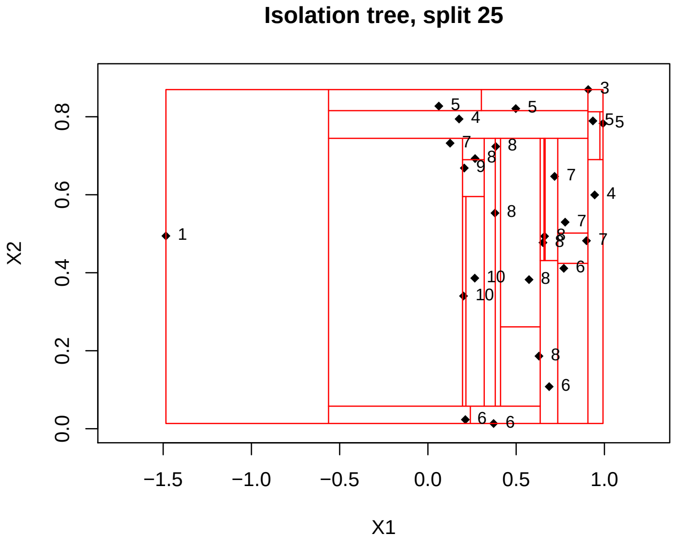
Step 2:



- The trick: an anomaly should be isolated faster than normal data.

Functional Isolation Forest

Illustration: Isolation tree



Children node construction in a functional isolation tree

If a node (j, k) is non terminal, it is split in three steps as follows:

1. Choose a Split function $\mathbf{d}$ according to the probability distribution $\boldsymbol{\nu}$ on $\mathcal{D}$.

Children node construction in a functional isolation tree

If a node (j, k) is non terminal, it is split in three steps as follows:

1. Choose a Split function $\mathbf{d}$ according to the probability distribution $\boldsymbol{\nu}$ on $\mathcal{D}$.
2. Choose randomly and uniformly a Split value γ in the interval

$$\left[\min_{\mathbf{x}\in\mathcal{S}_{j,k}} \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}}, \max_{\mathbf{x}\in\mathcal{S}_{j,k}} \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}}\right],$$

Children node construction in a functional isolation tree

If a node (j, k) is non terminal, it is split in three steps as follows:

1. Choose a Split function $\mathbf{d}$ according to the probability distribution $\boldsymbol{\nu}$ on $\mathcal{D}$.
2. Choose randomly and uniformly a Split value γ in the interval

$$\left[\min_{\mathbf{x}\in\mathcal{S}_{j,k}} \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}}, \max_{\mathbf{x}\in\mathcal{S}_{j,k}} \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}}\right],$$

3. Form the children subsets

$$\begin{aligned} \mathcal{C}_{j+1,2k} &= \mathcal{C}_{j,k} \cap \{\mathbf{x} \in \mathcal{H} : \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}} \leq \gamma\}, \\ \mathcal{C}_{j+1,2k+1} &= \mathcal{C}_{j,k} \cap \{\mathbf{x} \in \mathcal{H} : \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}} > \gamma\}. \end{aligned}$$

   as well as the children training datasets

$$\mathcal{S}_{j+1,2k} = \mathcal{S}_{j,k} \cap \mathcal{C}_{j+1,2k} \text{ and } \mathcal{S}_{j+1,2k+1} = \mathcal{S}_{j,k} \cap \mathcal{C}_{j+1,2k+1}.$$

Children node construction in a functional isolation tree

If a node (j, k) is non terminal, it is split in three steps as follows:

1. Choose a Split function $\mathbf{d}$ according to the probability distribution $\boldsymbol{\nu}$ on $\mathcal{D}$.
2. Choose randomly and uniformly a Split value γ in the interval

$$\left[\min_{\mathbf{x}\in\mathcal{S}_{j,k}} \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}}, \max_{\mathbf{x}\in\mathcal{S}_{j,k}} \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}}\right],$$

3. Form the children subsets

$$\begin{aligned} \mathcal{C}_{j+1,2k} &= \mathcal{C}_{j,k} \cap \{\mathbf{x} \in \mathcal{H} :\ \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}} \leq \gamma\}, \\ \mathcal{C}_{j+1,2k+1} &= \mathcal{C}_{j,k} \cap \{\mathbf{x} \in \mathcal{H} :\ \langle \mathbf{x}, \mathbf{d}\rangle_{\mathcal{H}} > \gamma\}. \end{aligned}$$

   as well as the children training datasets

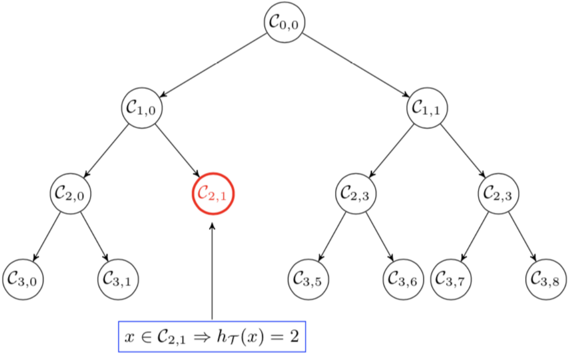
$$\mathcal{S}_{j+1,2k} = \mathcal{S}_{j,k} \cap \mathcal{C}_{j+1,2k} \text{ and } \mathcal{S}_{j+1,2k+1} = \mathcal{S}_{j,k} \cap \mathcal{C}_{j+1,2k+1}.$$

Stop when only one observation is in each node: isolation.

Anomaly score prediction

- One may then define the piecewise constant function $h_\tau : \mathcal{H} \to \mathbb{N}$ by: $\forall \boldsymbol{x} \in \mathcal{H}$,

  $h_\tau(\boldsymbol{x}) = j$ if and only if $x \in \mathcal{C}_{j,k}$ and $\mathcal{C}_{j,k}$ is associated to a terminal node.



Anomaly score prediction

Anomaly score calculation for observation $\boldsymbol{x}$:

1. For each isolation tree $i \in \{1, ..., N\}$, locate $\boldsymbol{x}$ in a terminal node and calculate the depth of this node $h_i(\boldsymbol{x})$.
2. Attribute the anomaly score:

$$s_n(\boldsymbol{x}) = 2^{-\frac{1}{N \cdot c(n)} \sum_{i=1}^{N} h_i(\boldsymbol{x})},$$

with $c(n) = 2H(n-1) - \frac{2(n-1)}{n}$ where $H(k)$ is the harmonic number and can be estimated by $\ln(k) + 0.5772156649$.

Anomaly score prediction

Anomaly score calculation for observation $\boldsymbol{x}$:

1. For each isolation tree $i \in \{1, ..., N\}$, locate $\boldsymbol{x}$ in a terminal node and calculate the depth of this node $h_i(\boldsymbol{x})$.
2. Attribute the anomaly score:

$$s_n(\boldsymbol{x}) = 2^{-\frac{1}{N \cdot c(n)} \sum_{i=1}^{N} h_i(\boldsymbol{x})},$$

with $c(n) = 2H(n-1) - \frac{2(n-1)}{n}$ where $H(k)$ is the harmonic number and can be estimated by $\ln(k) + 0.5772156649$.

Score behavior:

- when $\frac{1}{N}\sum_{i=1}^{N} h_i(\boldsymbol{x}) \to c(n)$, $s_n(\boldsymbol{x}) \to 0.5$,
- when $\frac{1}{N}\sum_{i=1}^{N} h_i(\boldsymbol{x}) \to 0$, $s_n(\boldsymbol{x}) \to 1$,
- when $\frac{1}{N}\sum_{i=1}^{N} h_i(\boldsymbol{x}) \to n-1$, $s_n(\boldsymbol{x}) \to 0$.

Contents

Parameters of FIF

- Classical parameters of Isolation Forest :
  - The number of trees, the size of the subsample and the height limit.

- New parameters due to the functional setup :

1. The dictionary $\mathcal{D}$.
2. The probability measure $\boldsymbol{\nu}$.
3. The scalar product $\langle ., . \rangle_{\mathcal{H}}$.

The role of the scalar product

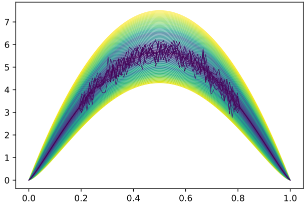
- Compromise between both location and shape :

$$\langle \mathbf{f}, \mathbf{g} \rangle := \alpha \times \frac{\langle \mathbf{f}, \mathbf{g} \rangle_{L_2}}{||\mathbf{f}|| \, ||\mathbf{g}||} + (1 - \alpha) \times \frac{\langle \mathbf{f}', \mathbf{g}' \rangle_{L_2}}{||\mathbf{f}'|| \, ||\mathbf{g}'||}, \quad \alpha \in [0, 1] \, ,$$

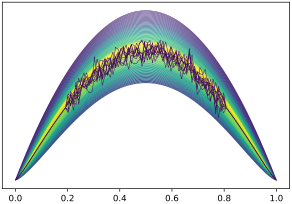
Example on a toy dataset :

- 90 curves defined by $\mathbf{x}(t) = 30(1 - t)^q t^q$ with q equispaced in $[1, 1.4]$,
- 10 *abnormal* curves defined by $\mathbf{x}(t) = 30(1 - t)^{1.2} t^{1.2}$ noised by $\varepsilon \sim \mathcal{N}(0, 0.3^2)$ on the interval $[0.2, 0.8]$.

$\alpha = 1$

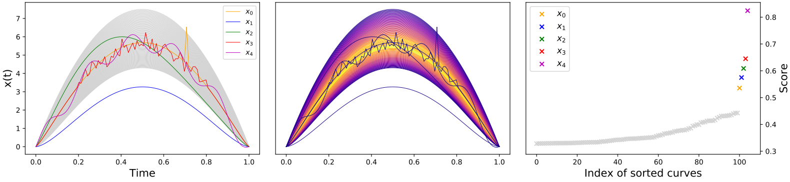


$\alpha = 0$



Ability to detect a variety of anomalies

- Sobolev inner product: $\langle .,. \rangle_{W_{1,2}}$.
- Gaussian wavelets dictionary $\mathbf{d}_{\theta,\sigma}(t) = \frac{2}{\sqrt{3\sigma}\pi^{1/4}} \left(1 - \left(\frac{t-\theta}{\sigma}\right)^2\right) \exp\left(\frac{-(t-\theta)^2}{2\sigma^2}\right)$.
- Uniform measure $\boldsymbol{\nu}$.



Contents

Performance on real datasets (1)

- FIF with 4 setups (Dictionary+scalar product):
  - Dyadic indicator (DI)+L_2
  - Cosine (Cos)+L_2
  - Cosine (Cos)+Sobolev
  - Dataset itself (Self)+L_2

  Competitors:

- *Isolation Forest*,*Local Outlier Factor*, *One-class SVM* after dimension reduction by FPCA.
- fHD_{RP} : Random projection method with functional Halspace depth.
- fSDO : Functional Stahel-Donoho Outlyingness.

Performance on real datasets (2)

Methods :	DI_{L_2}	Cos_{Sob}	Cos_{L_2}	$Self_{L_2}$	IF	LOF	OCSVM	fHD_{RP}	fSDO
Chinatown	0.93	0.82	0.74	0.77	0.69	0.68	0.70	0.76	0.98
Coffee	0.76	0.87	0.73	0.77	0.60	0.51	0.59	0.74	0.67
ECGFiveDays	0.78	0.75	0.81	0.56	0.81	0.89	0.90	0.60	0.81
ECG200	0.86	0.88	0.88	0.87	0.80	0.80	0.79	0.85	0.86
Handoutlines	0.73	0.76	0.73	0.72	0.68	0.61	0.71	0.73	0.76
SonyRobotAI1	0.89	0.80	0.85	0.83	0.79	0.69	0.74	0.83	0.94
SonyRobotAI2	0.77	0.75	0.79	0.92	0.86	0.78	0.80	0.86	0.81
StarLightCurves	0.82	0.81	0.76	0.86	0.76	0.72	0.77	0.77	0.85
TwoLeadECG	0.71	0.61	0.61	0.56	0.71	0.63	0.71	0.65	0.69
Yoga	0.62	0.54	0.60	0.58	0.57	0.52	0.59	0.55	0.55
EOGHorizontal	0.72	0.76	0.81	0.74	0.70	0.69	0.74	0.73	0.75
CinECGTorso	0.70	0.92	0.86	0.43	0.51	0.46	0.41	0.64	0.80
ECG5000	0.93	0.98	0.98	0.95	0.96	0.93	0.95	0.91	0.93

Table: AUC of different anomaly detection methods calculated on the test set. Bold numbers correspond to the best result.

Contents

Extension to multivariate functional data

FIF can be easily extended to the multivariate functional data, *i.e.* when the quantity of interest lies in $\mathbb{R}^d$ for each moment of time:

$$\begin{aligned} x : [0,1] &\longrightarrow \mathbb{R}^d \\ t &\longmapsto \big((x^1(t),\ \dots,\ x^d(t)\big) \end{aligned}$$

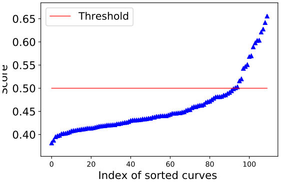
- Coordinate-wise sum of the d corresponding scalar products:

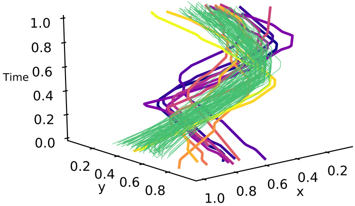
$$\langle \mathbf{f}, \mathbf{g} \rangle_{L_2^{\otimes d}} := \sum_{i=1}^{d} \langle f^{(i)}, g^{(i)} \rangle_{L_2}$$

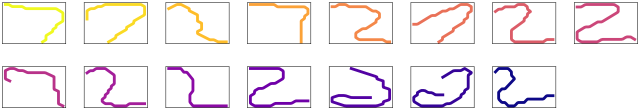
- Dictionaries : Composed by univariate function on each axis, multivariate wavelets, multivariate Brownian motion ...

Example with MNIST dataset

We extract the digits' contours and obtain bivariate functional curves from MNIST dataset. Each digit is transformed into a curve in $(L_2([0,1]) \times L_2([0,1]))$ using length parametrization on $[0,1]$.







Contents

Connection to data depth and supervised classification

- One may define a functional depth by $D_{FIF}(x; \mathcal{S}) = 1 - s_n(x; \mathcal{S})$.

Assume that we have a training classification dataset of q classes $\mathcal{S} = \mathcal{S}^1 \cup \ldots \cup \mathcal{S}^q$.

- Low dimensional representation based on depth-based map can be defined by

$$\mathbf{x} \mapsto \phi(\mathbf{x}) = \left(D_{FIF}(\mathbf{x}; \mathcal{S}^1), \ldots, D_{FIF}(\mathbf{x}; \mathcal{S}^q)\right) \in [0, 1]^q \,.$$

- One may define a DD-plot classifier by using a classifier on the low dimension representation of the functional dataset.

Example of depth map on MNIST dataset

$\mathcal{S}$ is constructed by taking 100 digits from class 1, 100 from class 5 and 100 from class 7.

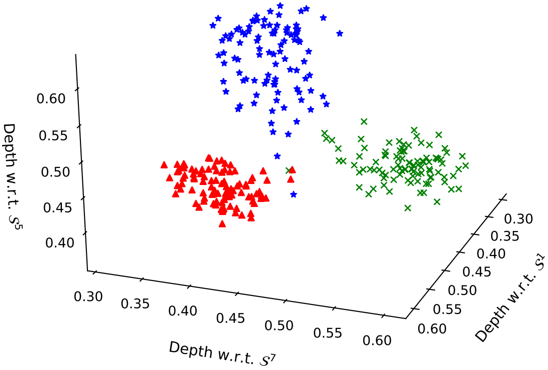


Figure: Depth space embedding of the three digits (1, 5 and 7) of the MNIST dataset.

Conclusion

- New anomaly detection algorithm for functional data:
  - Great flexibility but dictionaries (and scalar product) are tricky to choose in an unsupervised setting.
  - Low complexity and memory requierement.
- Lack of theoretical garanties!

Staerman, G., Mozharovskyi, P., Clémençon, S., and d'Alché-Buc, F. **Functional Isolation Forest.** *ACML 2019.*

All codes are available at:
https://github.com/guillaumestaermanML/FIF.

Thank you for your attention!

Questions?