Anomaly detection algorithms in Scikit-Learn

Nicolas Goix

Supervision: Alexandre Gramfort

Institut Mines-Télécom, Télécom ParisTech, CNRS-LTCI

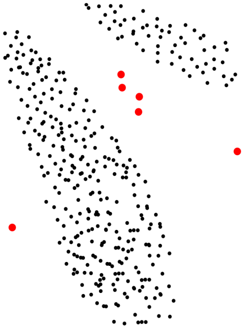
OSI day, Oct 2015

1. What is Anomaly Detection?
2. Isolation Forest algorithm
3. Examples

Anomaly Detection (AD)

What is Anomaly Detection ?

”Finding patterns in the data that do not conform to expected behavior”



Huge number of applications: Network intrusions, credit card fraud detection, insurance, finance, military surveillance,...

Machine Learning context

Different kind of Anomaly Detection

- **Supervised** AD
  - Labels available for both normal data and anomalies
  - Similar to rare class mining
- **Semi-supervised** AD (Novelty Detection)
  - Only normal data available to train
  - The algorithm learns on normal data only
- **Unsupervised** AD (Outlier Detection)
  - no labels, training set = normal + abnormal data
  - Assumption: anomalies are very rare

Important litterature in Anomaly Detection:

- **statistical AD techniques**
  fit a statistical model for normal behavior
  ex: EllipticEnvelope
- **density-based**
  - ex: Local Outlier Factor (LOF) and variantes (COF ODIN LOCI)
- **Support estimation** - OneClassSVM - MV-set estimate
- **high-dimensional techniques:** - Spectral Techniques - Random Forest - Isolation Forest

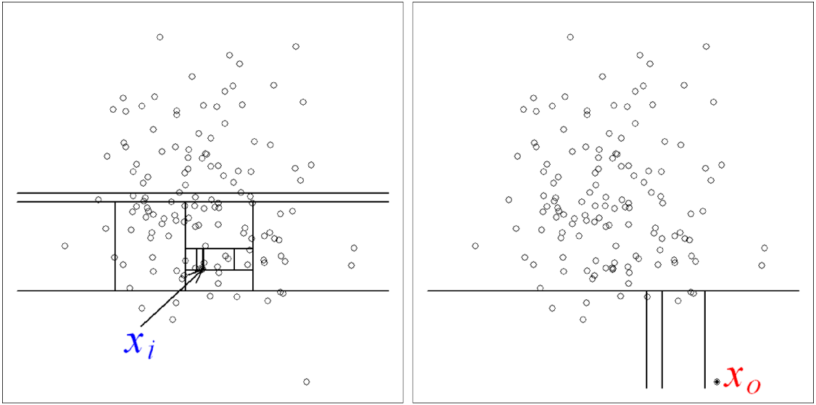
1. What is Anomaly Detection?

2. Isolation Forest algorithm

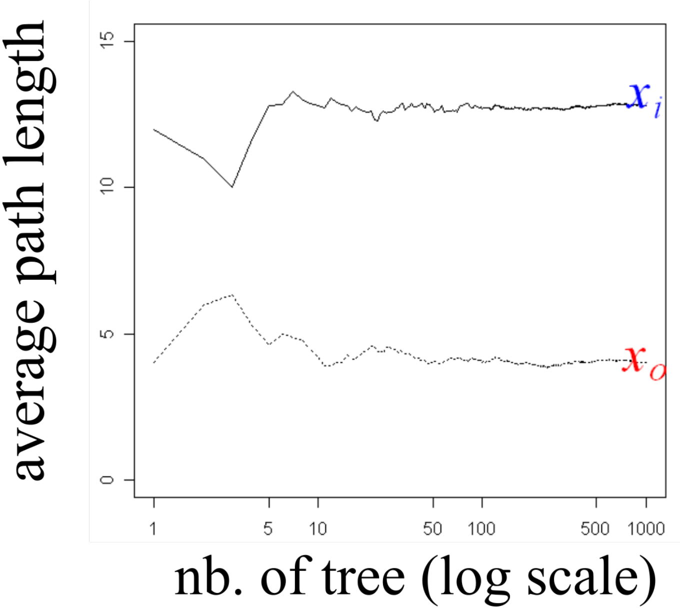
3. Examples

Idea:

Liu Tink Zhou icdm2008



(a) Isolating x_i (b) Isolating x_o


average path length
15
10
5
0
x_i
x_o
1
5
10
50
100
500
1000
nb. of tree (log scale)

IsolationForest.fit(X)

IsolationForest

Inputs: X, n_estimators, max_samples

Output: Forest with:

- # trees = n_estimators
- sub-sampling size = max_samples
- maximal depth $\mathit{max\_depth} = \mathit{int}(\log_2 \mathit{max\_samples})$

Complexity: O(n_estimators max_samples log(max_samples))

default: n_estimators=100, max_samples=256

IsolationForest.predict(X)

Finding the depth in each tree

```
depth(Tree, X):
    # - Finds the depth level of the leaf node
    #   for each sample x in X.
    # - Add average_path_length(n_samples_in_leaf)
    #   if x not isolated
```

$$score(x, n) = 2^{-\frac{E(depth(x))}{c(n)}}$$

Complexity: O(n_samples n_estimators log(max_samples))

1. What is Anomaly Detection?

2. Isolation Forest algorithm

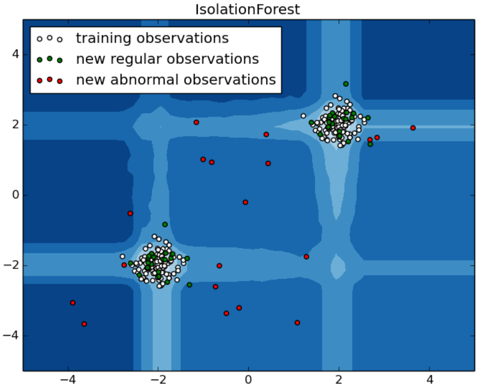
3. Examples

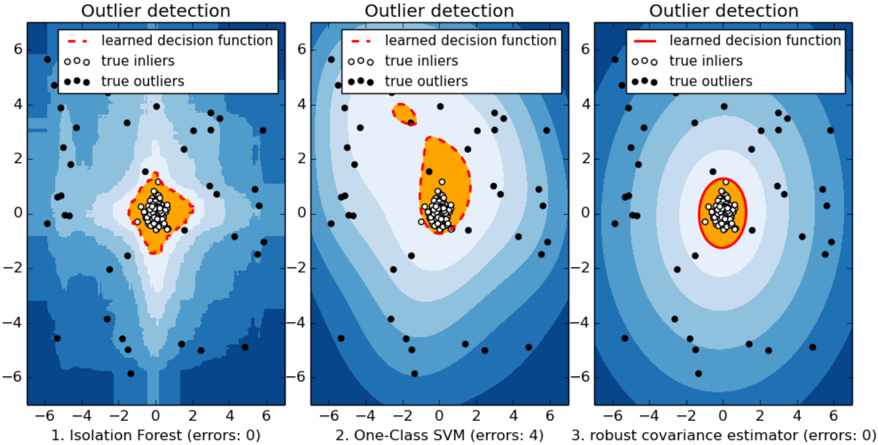
Examples

- code example:

```
>> from sklearn.ensemble import IsolationForest
>> IF = IsolationForest()
>> IF.fit(X_train)   # build the trees
>> IF.predict(X_test)   # find the average depth
```

- plotting decision function:




Outlier detection

Outlier detection

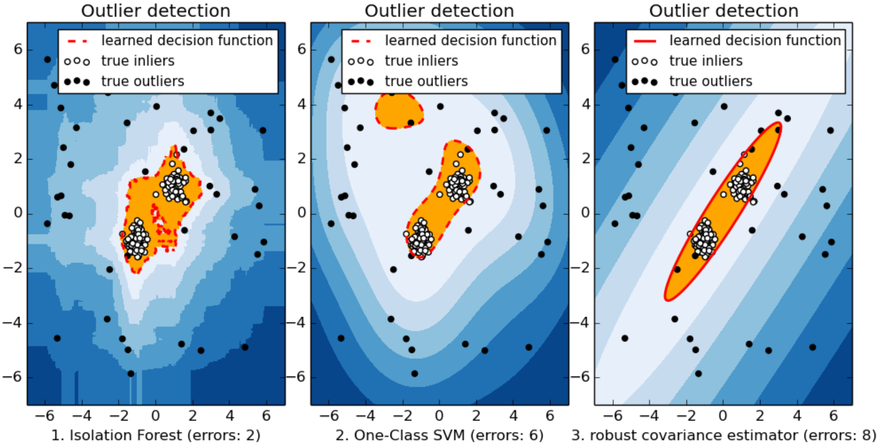
Outlier detection

1. Isolation Forest (errors: 0)

2. One-Class SVM (errors: 4)

3. robust covariance estimator (errors: 0)

```
n_samples_normal = 150
n_samples_outiers = 50
```



Outlier detection

Outlier detection

Outlier detection

1. Isolation Forest (errors: 2)

2. One-Class SVM (errors: 6)

3. robust covariance estimator (errors: 8)

n_samples_normal = 150
n_samples_outiers = 50

Thanks !