Détection des anomalies

Isolation Forest

Ricco Rakotomalala

Université Lumière Lyon 2

Plan

1. Problématique
2. Isolation Forest
3. Un exemple
4. Conclusion
5. Références

Problématique

DÉTECTION DES ANOMALIES

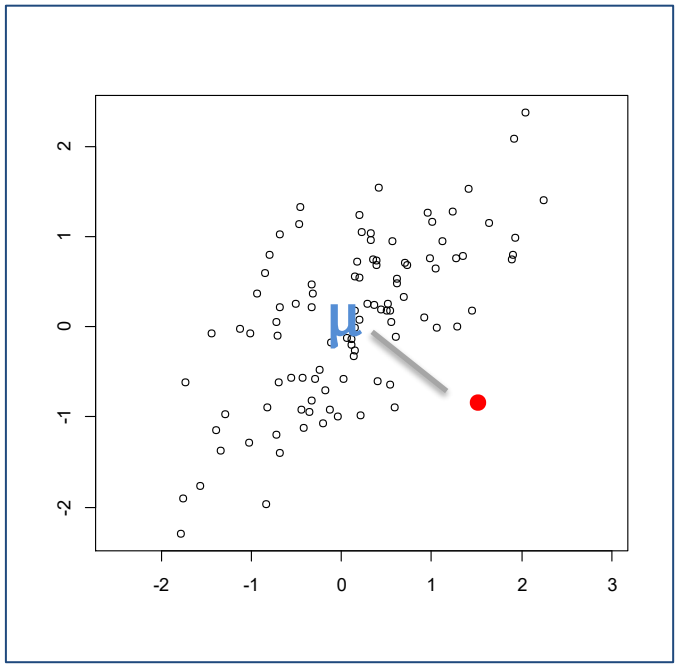
Anomalie ?

Détection des points atypiques (outlier detection) : un ou des points s'écartent significativement des autres dans une base de données. Ils sont épars et localisés dans une zone peu dense des données (s'ils forment un groupe compact, on ne peut pas vraiment parler d'anomalies)

Contexte multivarié : un point atypique s'écarte des autres au regard de l'ensemble des variables actives c.-à-d. il peut ne pas être atypique au sens de chaque variable (ex. taille = 185 cm, poids = 50 kg) mais le devient quand on appréhende les variables simultanément.

Approche simple – Distance de Mahalanobis

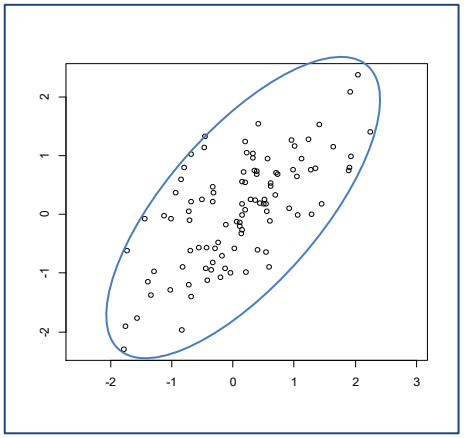
Pour chaque point, calculer la distance par rapport au barycentre (μ) – qui sert de référence – en tenant compte de la forme du nuage de points (via la covariance Σ).



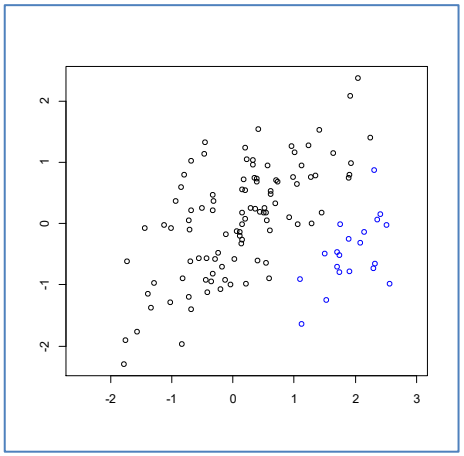
$$d^2_{(\mu,\Sigma)}(x_i) = (x_i - \mu)'\Sigma^{-1}(x_i - \mu)$$

Le point rouge n'est pas atypique sur les deux axes pris individuellement, mais l'est par rapport à la forme du nuage de points

Problème distance de Mahalanobis



Les calculs de μ et Σ peuvent être affectés par les points atypiques [c'est un comble, on essaie de les identifier justement] (Remarque : des solutions robustes existent… ex. « Minimum Covariance Determinant estimator », Rousseuw, 1984)



Les données peuvent être non-gaussiens, ou clustérisées, le barycentre global ne veut plus rien dire.

Identification des points atypiques à l'aide d'une forêt d'arbres

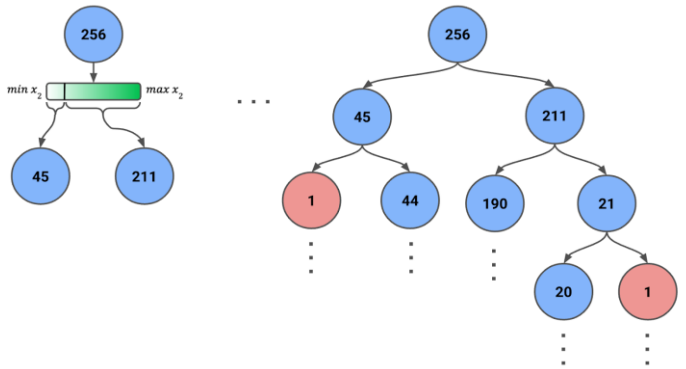
ISOLATION FOREST

Principe de l'algorithme Isolation Forest

Créer une multitude (n_estimators) d'arbres de décision (sans variable cible, sans critère à optimiser) au hasard (choix de variables [toutes quantitatives], seuil de découpage), les points suspicieux sont ceux « facilement » isolés des autres dans les feuilles des arbres.

Algorithme (pour un arbre)

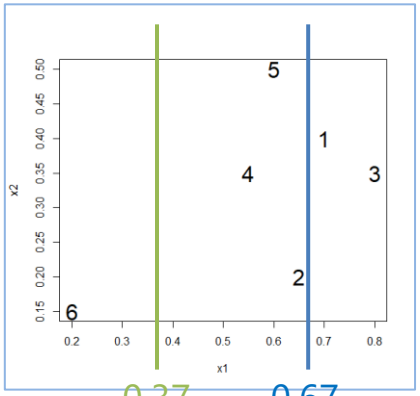
1. Sélectionner au hasard une variable
2. Choisir au hasard un point de coupure
3. Segmenter les données en 2 sous-groupes
4. Répéter itérativement le processus jusqu'à :
   a. Un nœud ne contenant qu'un individu devient une feuille
   b. Un nœud atteint une profondeur [nombre de niveaux] prédéfinie (ex. scikit-learn : $\log_2(n)$ où n = nombre d'observations ; ou autre…)

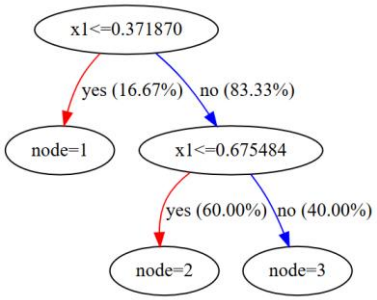


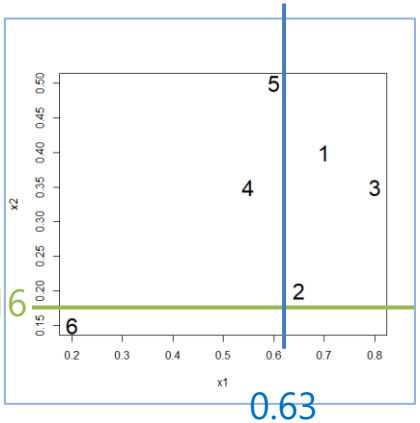
https://www.datacamp.com/tutorial/isolation-forest

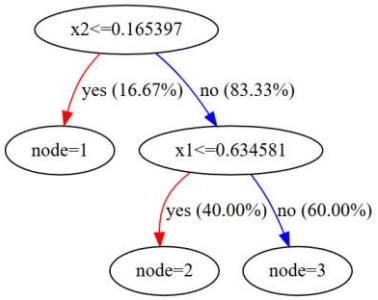
⇨ Le processus est répété (n_estimators) fois.

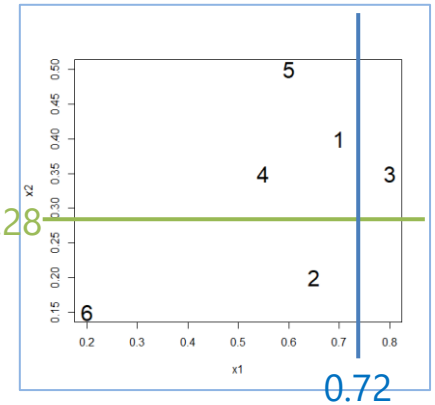
Un exemple dans le plan

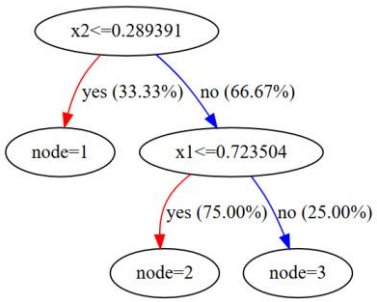












Paramètres (pour rendre lisibles les résultats) :

- max_depth = 2
- n_estimators = 3

(a) Le point n°6 est vite isolé

(b) Plus généralement, un indicateur pour identifier le caractère atypique d'un point est la profondeur des feuilles dans laquelle il se situe « en moyenne ».

Quelques définitions et calculs

$h(x)$ Longueur du chemin (niveau du sommet par rapport à la racine [niveau 0]) jusqu'à la feuille dans laquelle se situe l'observation x

$E[h(x)]$ Longueur moyenne du chemin pour x sur l'ensemble des arbres.

➔ Plus elle est faible, plus l'observations est « suspecte »

Problème : plus le nombre d'observations « n » est élevé, plus les arbres ont tendance à être grands. Il faut relativiser le résultat.

$c(n)$ Est la longueur moyenne des chemins dans un arbre binaire pour un échantillon de taille « n » avec :

$$c(n) = 2 \times H(n-1) - \frac{2 \times (n-1)}{n}$$

Où $H(n-1) = \sum_{k=1}^{n-1} \frac{1}{k}$

Quelques définitions et calculs (suite)

Nous obtenons ainsi un « score » normalisé (par rapport à la taille de l'échantillon), compris entre 0 et 1, avec :

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

⇨ Un point est atypique si E[h(x)] << c(n) et s → 1

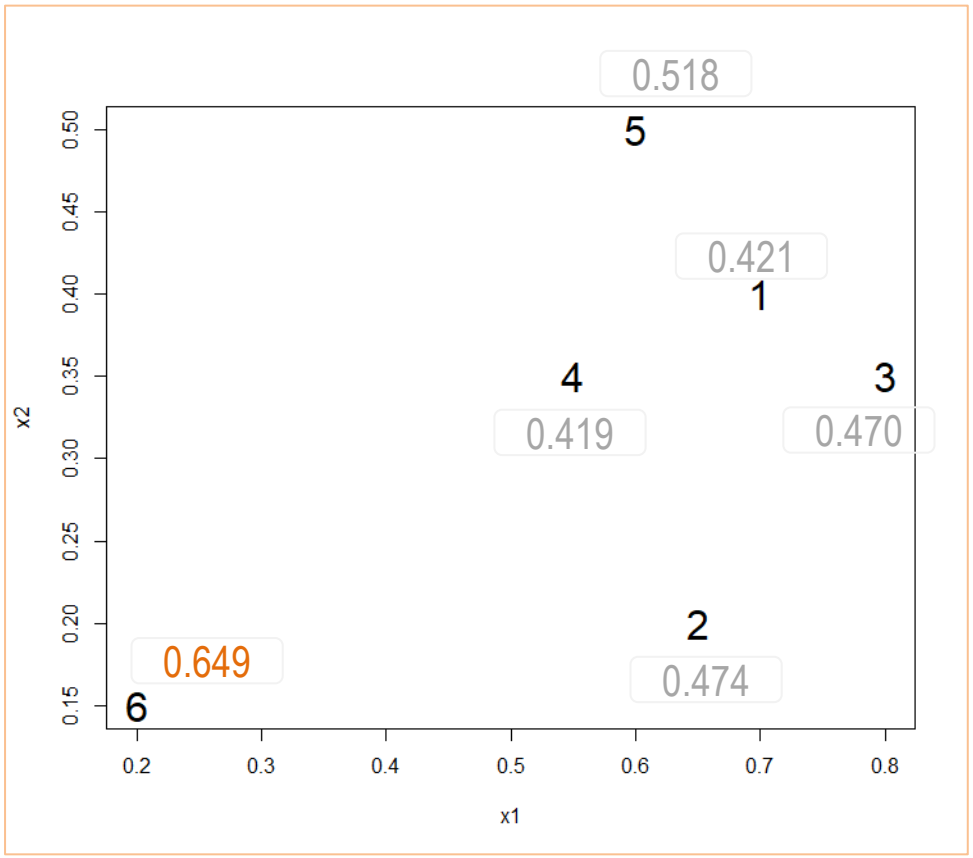
⇨ Un point est au centre d'un zone dense si E[h(x)] >> c(n) et s → 0

Remarques :

- Certaines librairies utilisent d'autres types de normalisation et de sens de variation (→ 0 : atypique [outlier], → 1 : normal [inlier] ; ou encore des valeurs négatives…)
- Il n'y a pas vraiment de valeur seuil universelle
- Certaines librairies introduisent un « taux de contamination » (pourcentage de points) pour identifier les observations suspectes (ex. scikit-learn [Python]), difficile à fixer
- Le mieux est d'identifier les « décrochages » dans le graphique des valeurs

Pour notre exemple dans le plan

Librairie « isotree » pour R (ntrees = 100, max_depth = 3)



ISOLATION FOREST – Avantages et inconvénients

\+

- Robuste à la présence de points atypiques (ouf !)
- Ne nécessite pas le calcul de distances
- (Et donc) Ne nécessite pas une normalisation préalable des données
- (+) Peut-être adaptée aux variables catégorielles
- (+) Possibilité de prendre en compte des combinaisons de variables pour les « splits »

\-

- Pas (peu) adaptée à la détection des nouveautés (novelty detection)
- Résultats peu interprétables (ex. variables responsables de l'écart…)
- Problème si anomalies = faible groupe de points proches (mais pas « cluster »)

!

- « Proportion » (taux de contamination) de points atypiques difficile à fixer :
- travailler plutôt sur les décrochages de valeurs des scores (loi du coude)

Identification des véhicules atypiques

UN EXEMPLE

Un ensemble de véhicules

Modele	Prix	Cylindree	Puissance	Poids	Conso
Daihatsu Cuore	11600	846	32	650	5.7
Suzuki Swift 1.0 GLS	12490	993	39	790	5.8
Fiat Panda Mambo L	10450	899	29	730	6.1
VW Polo 1.4 60	17140	1390	44	955	6.5
Opel Corsa 1.2i Eco	14825	1195	33	895	6.8
Subaru Vivio 4WD	13730	658	32	740	6.8
Toyota Corolla	19490	1331	55	1010	7.1
Ferrari 456 GT	285000	5474	325	1690	21.3
Mercedes S 600	183900	5987	300	2250	18.7
Maserati Ghibli GT	92500	2789	209	1485	14.5
Opel Astra 1.6i 16V	25000	1597	74	1080	7.4
Peugeot 306 XS 108	22350	1761	74	1100	9
Renault Safrane 2.2. V	36600	2165	101	1500	11.7
Seat Ibiza 2.0 GTI	22500	1983	85	1075	9.5
VW Golt 2.0 GTI	31580	1984	85	1155	9.5
Citroen ZX Volcane	28750	1998	89	1140	8.8
Fiat Tempra 1.6 Liberty	22600	1580	65	1080	9.3
Fort Escort 1.4i PT	20300	1390	54	1110	8.6
Honda Civic Joker 1.4	19900	1396	66	1140	7.7
Volvo 850 2.5	39800	2435	106	1370	10.8
Ford Fiesta 1.2 Zetec	19740	1242	55	940	6.6
Hyundai Sonata 3000	38990	2972	107	1400	11.7
Lancia K 3.0 LS	50800	2958	150	1550	11.9
Mazda Hachtback V	36200	2497	122	1330	10.8
Mitsubishi Galant	31990	1998	66	1300	7.6
Opel Omega 2.5i V6	47700	2496	125	1670	11.3
Peugeot 806 2.0	36950	1998	89	1560	10.8
Nissan Primera 2.0	26950	1997	92	1240	9.2
Seat Alhambra 2.0	36400	1984	85	1635	11.6
Toyota Previa salon	50900	2438	97	1800	12.8
Volvo 960 Kombi aut	49300	2473	125	1570	12.7

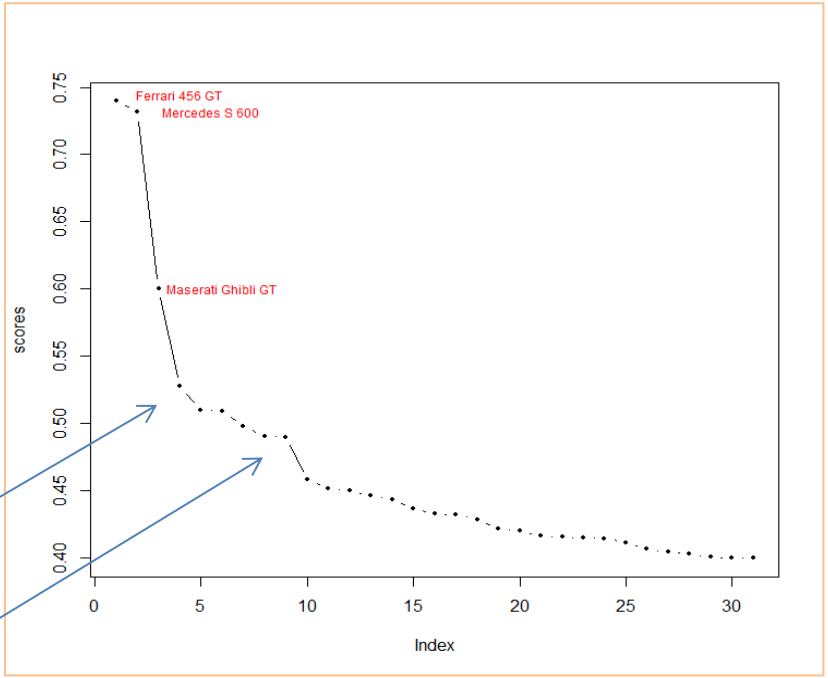
Comment identifier des véhicules atypiques – dont les caractéristiques se démarquent significativement des autres – dans cette base ?

Calculs sous R (1)

Les 3 premières, on comprend.

```
# chargement du fichier
library(xlsx)
cars <- read.xlsx("cars_outliers.xlsx",header=TRUE,sheetIndex=1)
rownames(cars) <- cars$Modele
cars <- cars[-1]
print(str(cars))

# identification des outliers
library(isotree)

# variables prises individuellement en split
iso <- isolation.forest(cars,ndim=1)

# calcul du score pour chaque obs.
scores <- predict.isolation_forest(iso,newdata=cars)
scores <- sort(scores,decreasing = TRUE)
print(round(scores,4))

# décroissance du score
plot(scores,type="b",pch=16,cex=0.5)
text(4,scores[1]+0.005,names(scores)[1],cex=0.75,col='red')
text(5.5,scores[2],names(scores)[2],cex=0.75,col='red')
text(6,scores[3],names(scores)[3],cex=0.75,col='red')
```

```
Ferrari 456 GT      Mercedes S 600      Maserati Ghibli GT
         0.7397              0.7317                  0.6005
    Daihatsu Cuore    Subaru Vivio 4WD      Fiat Panda Mambo L
         0.5275              0.5100                  0.5088
Toyota Previa salon  Suzuki Swift 1.0 GLS      Lancia K 3.0 LS
         0.4979              0.4902                  0.4898
Opel Corsa 1.2i Eco  Opel Omega 2.5i V6  Hyundai Sonata 3000
         0.4579              0.4514                  0.4496
Volvo 960 Kombi aut     VW Polo 1.4 60    Mazda Hachtback V
         0.4460              0.4435                  0.4368
Ford Fiesta 1.2 Zetec  Mitsubishi Galant  Seat Alhambra 2.0
         0.4325              0.4319                  0.4285
      Toyota Corolla  Fort Escort 1.4i PT       Volvo 850 2.5
         0.4214              0.4200                  0.4161
Opel Astra 1.6i 16V  Renault Safrane 2.2. V      Peugeot 806 2.0
         0.4152              0.4147                  0.4144
Honda Civic Joker 1.4 Fiat Tempra 1.6 Liberty  Seat Ibiza 2.0 GTI
         0.4114              0.4066                  0.4040
   Nissan Primera 2.0  Peugeot 306 XS 108       VW Golt 2.0 GTI
         0.4026              0.4004                  0.3999
Citroen ZX Volcane
         0.3996
```



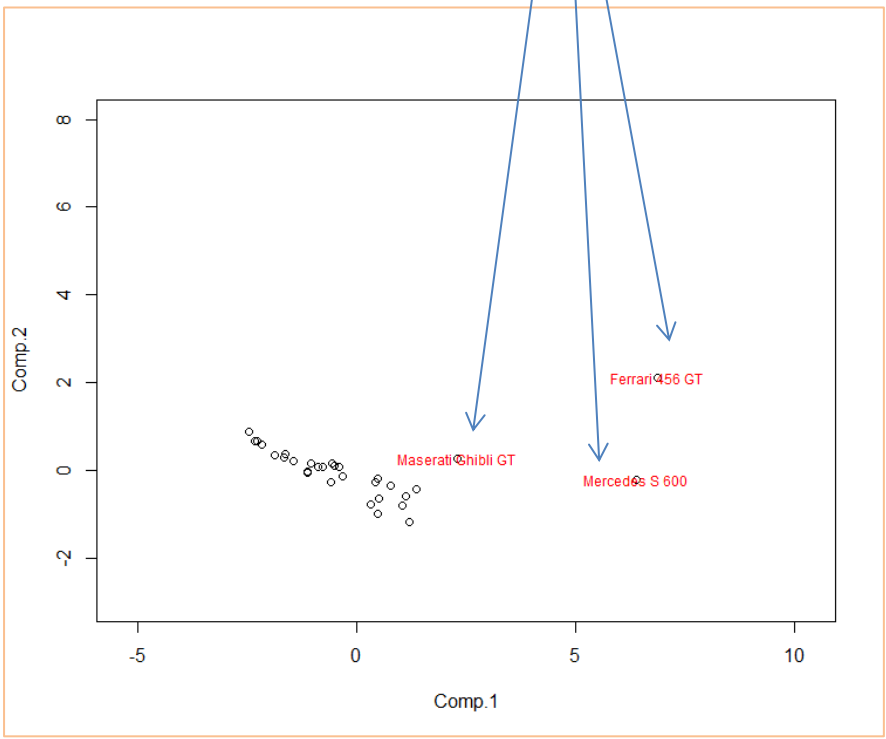
Le « coude » interpelle, forcément…

Là aussi peut-être ?

Calculs sous R (2)

```
# analyse en composantes principales
acp <- princomp(cars,cor=TRUE,scores=TRUE)
F1 <- acp$scores[,1]
F2 <- acp$scores[,2]

# projection dans le plan factoriel
plot(F1,F2,xlim=c(-3,8),ylim=c(-3,8),
            xlab='Comp.1',ylab='Comp.2',asp=1)
text(F1['Ferrari 456 GT'], F2['Ferrari 456 GT'],
            'Ferrari 456 GT',col='red',cex=0.75)
text(F1['Mercedes S 600'], F2['Mercedes S 600'],
            'Mercedes S 600',col='red',cex=0.75)
text(F1['Maserati Ghibli GT'], F2['Maserati Ghibli GT'],
            'Maserati Ghibli GT',col='red',cex=0.75)
```

L'analyse est confirmée



CONCLUSION

Isolation Forest - Conclusion

- La détection des anomalies a de nombreuses applications (identification des observations qui appartiennent à une autre population, des situations exceptionnelles, des comportements déviants [détection des intrusions par ex.], …)
- ISOLATION FOREST est une approche non-supervisée basée sur la construction aléatoire d'une multitude d'arbres
- Une observation isolée dans une feuille de la partie haute des arbres est suspecte lorsque la situation se répète
- L'identification du seuil ou du taux de contamination restent des problèmes ouverts…

RÉFÉRENCES

Références

- Wikipédia (en anglais) : « Outlier », « Anomaly detection ».
- Documentation Scikit-learn 1.8.0, « Novelty and Outlier Detection », section 2.7 / section 2.7.3.2.
- DataCamp, « Isolation Firest Guide : Explanation and Python Implementation ».
- David Cortes, « An Introduction to Isolation Forests », CRAN.