Universidad de Valparaíso CHILE

*Digital Object Identifier*

# Revisión del toolbox PYOD

**RODRIGO BARRERA**[1]
[1]Doctorado en Estadística, Universidad de Valparaíso, Valparaíso, Chile. (e-mail: rodrigo.barrerag@postgrado.uv.cl)

## ABSTRACT

La detección de valores atípicos -también llamados anómalos- hace referencia a la identificación de datos que se desvían de la distribución general. La presencia de valores atípicos puede deberse a múltiples razones; la cuestión es que la presencia de datos anómalos pueden causar problemas en el análisis estadístico. No existe una definición matemática rígida de lo que constituye una anomalía; en efecto, determinar si una observación es un valor atípico (o no) es, en última instancia, un ejercicio subjetivo. De todos modos, hoy se cuentan con métodos computacionales que son de utilidad para detectar la presencia de estas anomalías. La librería PyOD de Python cuenta con un amplio número de algoritmos para este propósito.

**INDEX TERMS** python;datos; atípicos;anomalías.

## I. INTRODUCCIÓN

Hay una gran cantidad de razones por las que existen valores atípicos. Una lista no exhaustiva es la siguiente: (1) errores en la entrada de datos, (2) presencia de un dispositivo no calibrado, (3) datos falsos. De forma general los valores atípicos se pueden clasificar en univariantes y multivariantes. Estos valores pueden afectar los resultados de los análisis y del modelado estadístico drásticamente, mas, los valores atípicos no deben considerarse como una cuestión negativa, como una cuestión a eliminar, sino que se debe poner esfuerzo en su detección.

Técnicas sencillas para detectar datos atípicos son los diagramas de caja y los gráficos de dispersión, sin embargo se requieren métodos más sofisticados y algoritmos dedicados para este propósito. Justamente da cuenta de lo anterior el toolbox PyOD de Python.

Aquí se estudiarán con más profundidad 3 algoritmos del toolbox PyOD. El criterio de inclusión guarda relación con la fecha de publicación del algoritmo, pues se privilegió los más recientes. Los algoritmos seleccionados son los que siguen: Rotation-based Outlier Detection [2]; Empirical Cumulative Distribution functions for Outlier Detection [4] y DeepSVDD [3].

### A. ESTADO DEL ARTE

Tal y como indican [9] el *toolbox* PyOD cuenta con las siguientes características: (a) solidez en su construcción, ya que se ejecutan pruebas periódicamente; (b) garantía de calidad. Lo anterior se respalda en que el proyecto sigue el estándar PEP8 [1]; la mantenibilidad está garantizada por CodeClimate[2], una herramienta automatizada de revisión de código y que entrega garantía de calidad. (c) Desarrollo basado en la comunidad y un repositorio con detallada documentación y ejemplos (Para más detalles consultar [10]).

Los modelos implementados en PyOD están supervisados por pruebas unitarias con integración continua entre plataformas, cobertura de código y controles de mantenimiento del mismo. La optimización se implementa toda vez que es posible: la compilación *just-in-time*[3] y la paralelización se habilitan en determinados modelos para la detección escalable de valores atípicos. PyOD es compatible con Python 2 y 3 en los principales sistemas operativos (Windows, Linux y MacOS).

En el contexto de aprendizaje de máquina existen tres (3) aproximaciones usuales para la detección de anomalías. Detección no supervisada, Detección de Novedades Semisupervisada (en inglés *Semi-supervised Novelty Detection*) y Clasificación de Anomalías supervisada. La librería PyOD solo se ocupa de los dos primeras.

*Detección no supervisada*

Los datos de entrenamiento contienen observaciones normales y anómalas, sin embargo no se caracterizan como tales. Los valores atípicos son identificados durante el proceso de ajuste. Aquellas observaciones que no

[1]PEP8 es una guía que indica las convenciones estilísticas a seguir para escribir código Python. Se trata de un conjunto de recomendaciones cuyo objetivo es ayudar a escribir código más legible

[2]Para más detalles consultar https://codeclimate.com

[3]Para una discusión detallada de este asunto véase [5]

pertenezcan a regiones de alta densidad se considerarán valores atípicos.

La detección de anomalías puede examinar grandes cantidades de datos para identificar *clusters* o grupos homogéneos en los que hay registros similares. Como se mencionó con antelación, los métodos tradicionales de identificación de valores atípicos examinan una o dos variables a la vez (utilizando diagramas de dispersión, por ejemplo).

Como resumen, la detección de anomalías no supervisada tiene por objeto detectar eventos extraños sin ningún conocimiento previo sobre los mismos. Usualmente se debe indicar el porcentaje de datos atípicos que contiene un conjunto de datos.

*Semi-supervised Novelty Detection*

Los datos de entrenamiento consisten únicamente en observaciones que describen el comportamiento no-anómalo. Los valores atípicos se definen como puntos que difieren de la distribución de los datos de entrenamiento. Aquellas nuevas observaciones que difieren de los datos de entrenamiento dentro de cierto umbral, incluso si forman una región de alta densidad, se consideran valores atípicos.

## II. MATERIALES Y MÉTODOS

PyOD es un conjunto de herramientas de Python -de código abierto- que tiene por objeto la detección de valores atípicos. Este *toolbox* se caracteriza por ser el único que proporciona acceso a una amplia gama de algoritmos de detección de valores atípicos. Incorpora además enfoques más recientes basados en redes neuronales, bajo una única y extremadamente bien documentada API.

A continuación se explica con mayor especificidad cada uno de los algoritmos seleccionados.

ROD (Rotation-based Outlier Detection) [2020] [Basado en proximidad]. Algoritmo de aprendizaje de valores atípicos novedoso y eficaz que se basa en la descomposición del espacio de atributos completos en diferentes combinaciones de subespacios, en los que los vectores 3D, que representan los puntos de datos por subespacios, son rotados sobre la mediana geométrica [4] , utilizando la fórmula de rotación de Rodrigues [5], para construir la puntuación periférica general.

La propuesta del algoritmo ROD es que el espacio de atributos completo se descompone en diferentes combinaciones de subespacios en que los vectores 3D, que representan los puntos de datos por subespacio 3D, se giran alrededor de la mediana geométrica dos veces en sentido contrario a las agujas del reloj usando la fórmula de rotación de Rodrigues. Los resultados de la rotación son paralelepípedos donde sus volúmenes son matemáticamente analizados como funciones de costo. Posteriormente, las particiones periféricas del espacio completo son reconstruidas tomando el promedio de los puntajes periféricos de todos los subespacios 3D.

ECOD (Empirical Cumulative Distribution functions for Outlier Detection)[2022].

ECOD usa información sobre la distribución de datos para determinar dónde la aparición de datos es menos probable. En particular, la función de distribución acumulativa empírica (ECDF) se estima individualmente para cada variable. Para determinar valores atípicos , ECOD utiliza la ECDF univariante para calcular las probabilidades marginales de cada variable y multiplicarlas. Este cálculo se realiza en espacio logarítmico, considerando las colas.

Para caracterizar de buena forma el procedimiento ECOD considere lo siguiente. Si se construye un histograma de los datos, capturando la información sobre la distribución, entonces se podrían utilizar los 'bins' con bajas frecuencias para determinar dónde están -posiblemente- los valores atípicos. El gráfico que se muestra a continuación ilustra justamente esa idea, a saber; los valores atípicos son puntos que se encuentran en partes de baja densidad de la distribución de probabilidad. Si la distribución es unimodal, las observaciones anómalas se encontraran en las colas de la distribución.

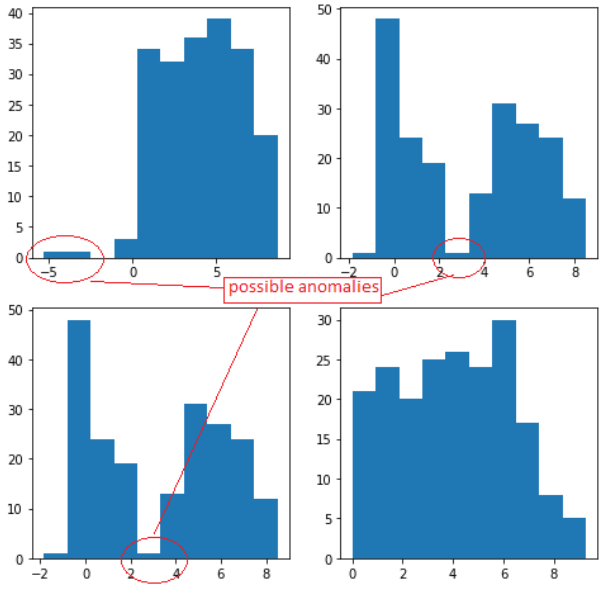


**FIGURE 1.** Caracterización del algoritmo ECOD [8]

ECOD utiliza justamente esta idea. Para cada observación, ECOD puntúa los valores atípicos según la probabilidad

[4]La mediana geométrica de un conjunto discreto de puntos de muestra en un espacio euclidiano es el punto que minimiza la suma de distancias a los puntos de muestra. Esto generaliza la mediana, que tiene la propiedad de minimizar la suma de distancias para los datos unidimensionales, y proporciona una tendencia central en dimensiones superiores.

[5]En la teoría del grupo de rotación SO(3), la fórmula de rotación de Rodrigues, que lleva el nombre de Olinde Rodrigues (1795-1851), es un algoritmo que permite rotar un vector en el espacio, dado un eje y un ángulo de rotación.

de observar un punto al menos tan 'extremo' como la observación dada, en términos de probabilidades de cola.

*DeepSVDD*

Naturalmente DeepSVDD tiene como base al algoritmo Support Vector Data Description (en adelante SVDD). La aplicación de SVDD para la detección de anomalías opera -en términos generales- como sigue: se ajusta la esfera más pequeña posible alrededor de los puntos de datos dados, lo que permite excluir algunos puntos como valores atípicos. La exclusión o no de un punto, se rige por una variable de holgura. Sobre esto último [7] argumentan que las variables de holgura de SVDD carecen de un sentido geométrico claro.

En aplicaciones prácticas, el conjunto de datos objetivo suele contener más de una clase de objetos y cada clase de objetos debe describirse y distinguirse simultáneamente. No obstante; el algoritmo SVDD sólo puede dar una descripción para el conjunto de datos objetivo, sin tener en cuenta las diferencias entre las diferentes clases de objetos, lo que supone una desventaja.

Por su parte, el algoritmo DeepSVDD entrena una red neuronal minimizando el volumen de una hiper-esfera que encierra las representaciones de la red de los datos, obligando a la red a extraer los factores comunes de variación. De forma similar a análisis de componentes principales, DeepSVDD podría utilizarse para detectar objetos periféricos en los datos calculando la distancia desde el centro.

Al igual que el algoritmo SVDD, DeepSVDD tiene como objetivo encontrar una hiper-esfera en el espacio de características. Sin embargo, la diferencia radica en el uso de un procedimiento de transformación de datos más profundo basado en la red neuronal profunda. $\Phi(x, W)$ representa los datos mapeados dados por la red $\Phi$ con parámetros $W$ . Para minimizar el volumen de la hiper-esfera que rodea los datos no-anómalos se define -bajo ciertas condiciones- el objetivo :

$$min_W \frac{1}{n}\sum_{i=1}^{n} max \parallel \Phi(x,W)) - c \parallel^2 + \frac{\lambda}{2}\sum \parallel W \parallel_F^2 \quad (1)$$

Las aspectos matemáticos del algortimo son tratados en detalle en [13].

En resumen, SVDD y DeepSVDD tienen una metodología similar. Sin embargo difieren en cuanto al espacio de optimización. Esto se puede apreciar gráficamente en la siguiente figura:

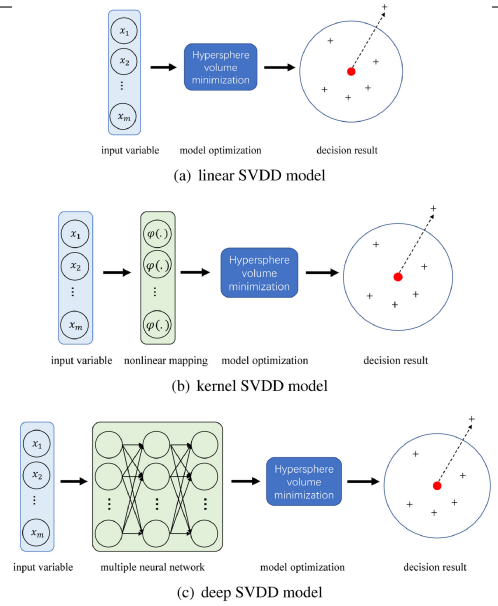


**FIGURE 2.** "Comparación de SVDD y DeepSVDD"

El algoritmo DeepSVDD es particularmente útil en varias áreas. Por ejemplo, protegiendo sitios de posibles amenazas. Detalles de esta aplicación se encuentran en [11]. Además se ha mostrado su utilidad en el análisis de imágenes, con aplicaciones a OCR (Optical Character Recognition). Detalles de esto aparecen en [12].

## REFERENCES


[1] (2022, Mayo 5) Welcome to PyOD documentation! Consultado en https://pyod.readthedocs.io/en/latest/.

[2] Almardeny, Y., Boujnah, N., Cleary, F. (2020). A novel outlier detection method for multivariate data. *IEEE Transactions on Knowledge and Data Engineering.*

[3] Zhang, Z., Deng, X. (2021). Anomaly detection using improved deep SVDD model with data structure preservation. *Pattern Recognition Letters*, 148, 1-6.

[4] Li, Z., Zhao, Y., Hu, X., Botta, N., Ionescu, C., Chen, G. H. (2022). ECOD: Unsupervised Outlier Detection Using Empirical Cumulative Distribution Functions. *arXiv preprint arXiv:2201.00382.*

[5] Just-in-time compilation (JIT). Computational Statistics in Python. (2022).Retrieved 24 May 2022, desde *https://people.duke.edu/ ccc14/sta-663-2016/18C_N umba.html*

[6] GitHub. 2022. use the Pyod for timeseries anomaly detection · Issue 9 · yzhao062/pyod. [online] Available at: <https://github.com/yzhao062/pyod/issues/9> [Accessed 25 May 2022].

[7] Pauwels, E.J., Ambekar, O. (2011). One Class Classification for Anomaly Detection: Support Vector Data Description Revisited. In: Perner, P. (eds) Advances in Data Mining. Applications and Theoretical Aspects. ICDM 2011. Lecture Notes in Computer Science(), vol 6870. Springer, Berlin, Heidelberg. *https://doi.org/10.1007/978-3-642-23184-1_3*

[8] Medium. 2022. Replace Outlier Detection by Simple Statistics with ECOD. [online] Available at: <https://medium.com/geekculture/replace-outlier-detection-by-simple-statistics-with-ecod-f95a7d982f79> [Accessed 25 May 2022].

[9] Zhao, Y., Nasrullah, Z., Li, Z. (2019). Pyod: A python toolbox for scalable outlier detection. *arXiv preprint arXiv:1901.01588.*

[10] Pyod.readthedocs.io. 2022. pyod 1.0.1 documentation. [online] Available at: <https://pyod.readthedocs.io/en/latest> [Accessed 29 May 2022].

[11] Moradi Vartouni, A., Shokri, M., Teshnehlab, M. (2021). Auto-Threshold Deep SVDD for Anomaly-based Web Application Firewall.

[12] Zhang, H., Davidson, I. (2021, March). Towards fair deep anomaly detection. In Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (pp. 138-148).

[13] Zhang, Zheng, and Xiaogang Deng. "Anomaly detection using improved deep SVDD model with data structure preservation." Pattern Recognition Letters 148 (2021): 1-6.