



UNIVERSIDAD DE GRANADA

INTELIGENCIA DE NEGOCIO

JANUARY 29, 2021

DetECCIÓN de anomalías

David Alberto Martín Vela

davidmv1996@correo.ugr.es
Doble Grado Ingeniería Informática y Matemáticas

Curso 2020-2021

Contents

Descripción y análisis del problema	2
Planteamiento del problema	3
Descripción de los algoritmos	6
Estudio experimental	9
Planteamiento de futuro	14
Referencias	15

Descripción y análisis del problema

¿Alguna vez nos hemos preguntado como los bancos detectan fraudes o en las redes sociales cuando sospechan que un inicio de sesión es fraudulento? Esto se realiza principalmente a través del proceso denominado Detección de anomalías (*Anomaly Detection*).

Una anomalía, por definición, es algo que se desvía de lo que es estándar, normal o esperado. La detección de anomalías o la detección de valores atípicos es el proceso de identificación de elementos raros, observaciones, patrones, valores atípicos o anomalías que diferirán significativamente de los elementos o patrones normales. Las anomalías a veces se denominan valores atípicos, novedades, ruido, desviaciones o excepciones.

Se dice que la información es poder, y cada vez se tiene más en cuenta que esa información en la sociedad actual viene dada por los datos. Ahora bien, una gran cantidad de datos conlleva poder si se manejan correctamente.

Según un artículo de Forbes [For] el **61%** de los vendedores planean usar el aprendizaje automático como parte de su estrategia de datos, dado que todavía hay empresas que se están perdiendo esta ventaja con el resto de los competidores. Se remarca el hecho de que, entre otros, pueden ayudar a descubrir palabras clave y otros elementos de las campañas de marketing que no se están aprovechando, prevenir las violaciones y amenazas a la seguridad y detectar amenazas y problemas antes de que causen daños.

En el mismo estudio de Forbes se menciona el caso de la empresa de consultoría Accenture. Casi el **10%** de sus 25 millones de procesos anuales de líneas de gastos estaban siendo marcados por incumplimiento o fraude. Mientras que su sistema basado en reglas funcionaba hasta cierto punto, Accenture implementó un algoritmo de aprendizaje automático para optimizar el proceso. Se utilizó para reducir los falsos positivos, detectar los valores anómalos y crear una solución no supervisada.

Por supuesto es difícil saber que pasa exactamente con los datos, pero es ahí donde entra la inteligencia artificial. Herramientas como por ejemplo Google Analytics, Facebook Ads y Shopify no son capaces de abordar todos los datos en grandes empresas. Y es aquí donde un negocio debe apostar por mecanismos de detección de anomalías con algoritmos de aprendizaje automático.

Al principio para orientar esta práctica alternativa a un tema específico, las primeras dos opciones que se me venían a la cabeza eran dos áreas del conocimiento, una primera opción, **la detección de terremotos**, debido a la gran cantidad de los mismos ocurrido últimamente en Granada [Ter], y otra opción, detección de anomalías en el campo de la bolsa, debido otro acontecimiento reciente donde **GameStop se**

dispara en bolsa tras la compra de acciones de usuarios de Reddit [Red].

Planteamiento del problema

Un resumen rápido de esta situación [pictoline] es que un grupo de inversores decidió apostar fijo por la caída de las acciones de GameStop sin arriesgarse, pues esta tienda llevaba perdiendo en bolsa desde bastante tiempo principalmente por el auge de las ventas digitales frente al formato físico. De esta forma, acordaron vender sus acciones a un precio fijo dentro de un periodo de tiempo determinado dando por sentado que las acciones de la tienda seguirían cayendo y deseando así que las acciones cayeran para obtener beneficio.

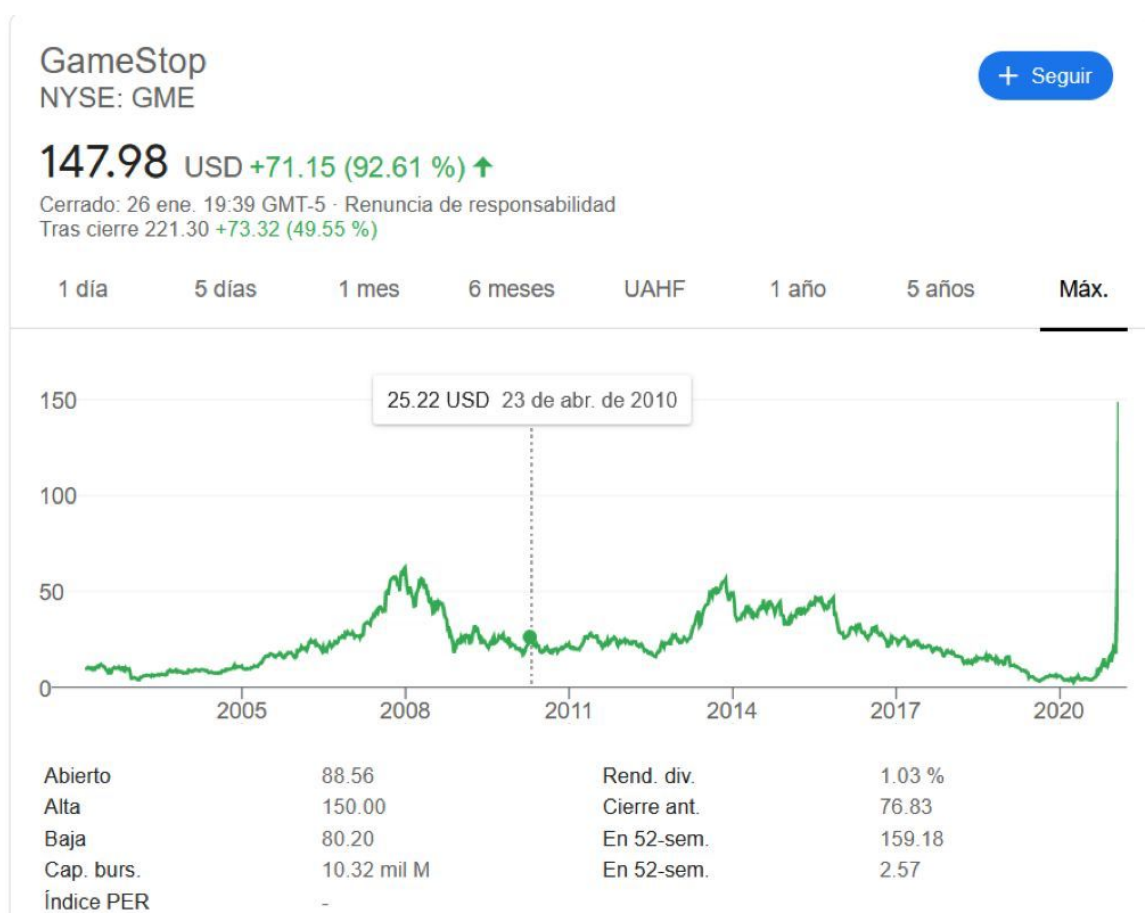


Figure 1: Gamstop stock from the last days

Sin embargo, esta estrategia llegó a oídos de los miembros del subreddit *wallstreetbets*, a quienes les pareció mal cómo se estaban comportando estos inversores. Decidieron tomar la decisión de comprar estas acciones baratas, inflando rápidamente el valor mucho más allá de lo que esperaban los administradores de fondos de cobertura, de manera que algunos miembros de Reddit han llegado a pagar miles de dólares con el objetivo de reventar los planes de los inversores mencionados anteriormente [1]. Mientras que *wallstreetbets* celebran la locura y dicen que no van a

vender y seguir comprando (incluso hay gente que compro hace año y medio una call con 50k y si la ejecuta se llevaría 36 millones ahora mismo) [Cal]. Ahora los compradores de Reddit deben calcular cuándo vender sus acciones para obtener beneficio, el cual podría ser de hasta 3.000 veces lo que compraron. Además, están explorando otros informes de **AMC** y **BlackBerry**, una cadena de cines estadounidense y una empresa de tecnología canadiense, para llevar a cabo acciones similares. **Otro tema interesante podría ser que debido a este fenómeno hay gente aplicando análisis de opiniones/sentimientos** en este foro de reddit para ver cuál puede ser el próximo objetivo pero esto ya se sale de nuestro tema elegido que es la detección de anomalías.

Para los datos, vamos a utilizar el paquete *pandas-datareader* [Pan]¹ donde extraeremos los datos trading volume data de Yahoo Finance. En nuestro caso, nuestras características de entrada serán una lista de símbolos ETF, los comentados anteriormente que corresponderán a Gamestop (GME), Blackberry (BB), Nokia (NOK) y AMC. Definiremos este entorno como nuestro "mercado", aunque en la práctica podríamos hacer que sea mucho, mucho más grande. Cogemos fechas desde hace 5 años hasta hoy, donde han ocurrido los acontecimientos recientes. Mostramos imágenes del trading volume y del precio de cierre en la figuras [2] y [3] respectivamente.

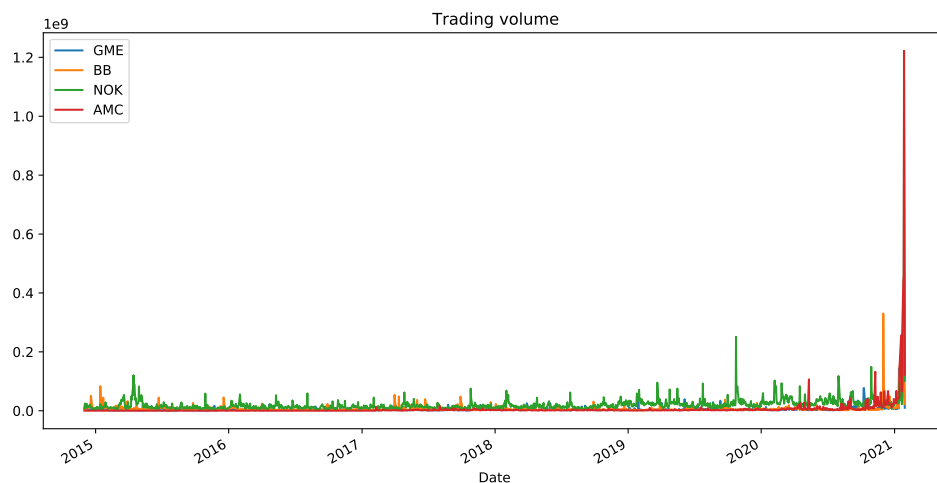


Figure 2: Trading volume data

¹The Pandas datareader is a sub package that allows one to create a dataframe from various internet datasources, currently including: Yahoo! Finance. Google Finance.

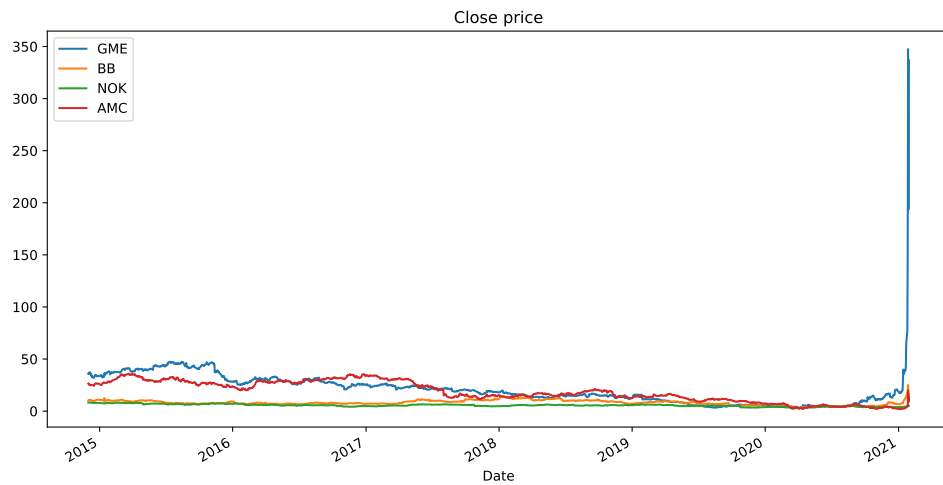


Figure 3: Closing Price

En el comercio como en la vida, a menudo es extremadamente valioso determinar si el entorno actual es anómalo o no de alguna manera. Si las cosas están actuando "normalmente", sabemos que nuestras estrategias pueden operar de cierta manera. Por ejemplo, si nos encontramos en un entorno comercial normal, podríamos emplear una estrategia de volatilidad en corto. Por otro lado, si identificamos que estamos en un mercado anormalmente emocionante, podría ser necesario emplear una estrategia que haga exactamente lo contrario: buscar oportunidades para el comercio basado en el impulso, por ejemplo. En ese tipo de mercado, acortar la volatilidad podría ser muy peligroso. El objetivo será aplicar una serie de algoritmos para determinar cuándo el volumen de operaciones de nuestra lista de símbolos se encuentra en un estado anómalo. Esto podría significar, por ejemplo, que estamos detectando un pico en el volumen de operaciones.

Descripción de los algoritmos

Vamos a utilizar y comparar algoritmos de las bibliotecas PyOD [Pyo] y Scikit-Learn Outlier Detection [Skl], primero, vamos a comentar algunos de ellos. El módulo Python Outlier Detection (PyOD) facilita el modelado de detección de anomalías. Recopila una amplia gama de técnicas que van desde el aprendizaje supervisado hasta las técnicas de aprendizaje no supervisado. No es necesario probar todas las técnicas para encontrar anomalías. Dependiendo de los datos, algunas técnicas funcionan mejor que otras. Típicamente el problema de detección de anomalías es un problema no supervisado, esto quiere decir que nuestros algoritmos no tienen etiquetas para entrenar. No obstante tenemos aproximaciones de algoritmos supervisados para este tipo de problemas, aunque debido al conjunto de datos que hemos elegido donde tenemos un problema de aprendizaje no supervisado (cluster) donde intentaremos aprender el patrón de los datos pero no mediante conjuntos de entrenamiento sino por esta demasiado lejos de un grupo, directamente sobre el conjunto aplicamos las técnicas, luego es un problema no supervisado donde nos centraremos principalmente en algoritmos no supervisados de las librerías comentadas anteriormente.

Los algoritmos que probaremos algoritmos no supervisados han sido escogidos de la documentación de PyOD [lista] y son los siguientes:

1. Linear Models for Outlier Detection

- (a) **PCA:** Principal Component Analysis es una reducción de dimensionalidad lineal que utiliza la descomposición de valores singulares de los datos para proyectarlo a un espacio dimensional inferior. Aunque tiene una gran cantidad de usos nosotros vamos a centrarnos en detección de anomalías. En este procedimiento, la matriz de covarianza de los datos se puede descomponer en vectores ortogonales, llamados autovectores, asociados con autovalores. los vectores propios con valores propios altos capturan la mayor parte de la varianza en los datos.

Por lo tanto, un hiperplano de baja dimensión construido por k autovectores puede capturar la mayor parte de la varianza en los datos. Sin embargo, los valores atípicos son diferentes de puntos de datos normales, que es más obvio en el hiperplano construido por los autovectores con pequeños autovalores.

Por lo tanto, los valores anómalos buscados se pueden obtener como la suma de los valores proyectados distancia de una muestra en todos los vectores propios y la puntuación será la suma de la distancia euclidiana

ponderada entre cada muestra y la hiperplano construido por los autovectores seleccionados.

- (b) **Minimum Covariance Determinant (MCD)** (use the mahalanobis distances as the outlier scores). Es un estimador robusto de covarianza.

Se aplicará el estimador de covarianza determinante de covarianza mínima en datos distribuidos en gauss, pero aún podría ser relevante en datos extraído de una distribución simétrica unimodal. No está destinado a ser utilizado con datos multimodales (el algoritmo utilizado para ajustar un objeto MinCovDet es probable que falle en tal caso). Se deben considerar métodos de búsqueda de proyecciones para hacer frente a multimodales conjuntos de datos.

Primero se ajusta un modelo determinante de covarianza mínima y luego calcule el Distancia de Mahalanobis como el grado atípico de los datos

- (c) **One-Class Support Vector Machines (OCSVM)**. Ya hemos visto en clase este tipo de modelos, solo que en este caso se plantea otra funcionalidad. Detección de valores atípicos sin supervisión esstimando el soporte de una distribución de alta dimensión. La implementación se basa en libsvm de Sklearn aunque utilizaremos el algoritmo de PyOD.

2. Proximity-Based Outlier Detection Models

- (a) **Local Outlier Factor (LOF)** La puntuación de anomalía de cada muestra se denomina Factor de valor atípico local (LOF). Mide la desviación local de densidad de una muestra dada con respecto a sus vecinos, es local en el sentido de que la puntuación de anomalía depende de qué tan aislado esté el objeto es con respecto al vecindario circundante. Más precisamente, la localidad está dada por k vecinos más cercanos, cuya distancia se utiliza para estimar la densidad local. Comparando la densidad local de una muestra con las densidades locales de sus vecinos, uno puede identificar muestras que tienen un sustancialmente menor densidad que sus vecinos. Estos se consideran valores atípicos.

- (b) **Clustering-Based Local Outlier Factor (CBLOF)** CBLOF toma como entrada el conjunto de datos y el modelo de clúster que se generado por un algoritmo de agrupamiento. Clasifica los grupos en pequeños clústeres y clústeres grandes utilizando los parámetros alfa y beta. Luego, la puntuación de anomalía se calcula en función del tamaño del clúster punto al que pertenece, así como la distancia al cúmulo grande más cercano.

Utiliza la ponderación para el factor de valores atípicos en función de los tamaños de los grupos como propuesto en la publicación original. Dado que esto puede llevar a comportamiento (no se encuentran valores atípicos cercanos a grupos pequeños), está deshabilitado Las puntuaciones de Outliers se calculan únicamente en función de su distancia a el centro grande más cercano.

De forma predeterminada, kMeans se utiliza para el algoritmo de agrupación en clústeres en lugar de Algoritmo Squeezer mencionado en el

artículo original por múltiples razones.

- (c) **k Nearest Neighbors (kNN)** De nuevo, ya hemos visto este tipo de algoritmo en la asignatura, solo que en este caso se usa la distancia al k th vecino más cercano como puntuación de anomalía.
- (d) **Median kNN Outlier Detection** Igual que antes solo que utilizando la median distance.
- (e) **Histogram-based Outlier Score (HBOS)** es un algoritmo es un eficiente sin supervisión. Asume la independencia de la función y calcula el grado de las anomalías mediante la construcción de histogramas.

3. Probabilistic Models for Outlier Detection

- (a) **Angle-Based Outlier Detection (ABOD)** Considera la relación entre cada punto y su (s) vecino (s). No considera las relaciones entre estos vecinos. La varianza de sus puntuaciones de coseno ponderadas a todos los vecinos podría verse como la puntuación periférica. Funciona bien en multidimensional data.

4. Outlier Ensembles and Combination Frameworks

- (a) **Isolation Forest** Utiliza la biblioteca scikit-learn internamente. En este método, la partición de datos se realiza mediante un conjunto de árboles. Isolation Forest proporciona una puntuación de anomalía al observar qué tan aislado está el punto en la estructura. Luego, la puntuación de anomalía se utiliza para identificar valores atípicos de observaciones normales Isolation Forest funciona bien en datos multidimensionales
- (b) **LSCP** LSCP es un conjunto de detección de valores atípicos paralelo no supervisado que selecciona detectores competentes en la región local de una instancia de prueba. Esta La implementación utiliza una estrategia de Promedio de Máximo (*Average of Maximum*). Primero, un heterogéneo La lista de detectores base se ajusta a los datos de entrenamiento y luego genera un la verdad pseudo-terrestre para cada instancia de tren es generada por tomando la puntuación máxima de valores atípicos.

Para cada instancia de prueba:

- i. La región local se define como el conjunto de puntos de entrenamiento más cercanos en subespacios de características muestreados aleatoriamente que ocurren con más frecuencia que un umbral definido en múltiples iteraciones.
- ii. Usando la región local, se define una verdad de pseudo terreno local y la La correlación de pearson se calcula entre el entrenamiento de cada detector base puntajes atípicos y la verdad pseudo fundamental.
- iii. Se construye un histograma a partir de las puntuaciones de correlación de Pearson; detectores en los contenedores más grandes se seleccionan como detectores de base competentes para el instancia de prueba.
- iv. Se toma la puntuación promedio de valores atípicos de los detectores competentes seleccionados para ser la puntuación final.

Estudio experimental

Todo lo relacionado con esta parte de la práctica puede encontrarse en mi repositorio de [github](#).

Vamos a proceder a aplicar los algoritmos anteriores con los datos comentado en la descripción del problema, en las figuras [2] y [3]. Inicialmente hagamos un describe para ver los datos en la tabla [1]

	GME	BB	NOK	AMC
count	1.047000e+03	1.047000e+03	1.047000e+03	1.047000e+03
mean	5.781856e+06	7.729197e+06	2.394326e+07	8.067518e+06
std	1.296132e+07	2.352192e+07	4.480880e+07	4.861418e+07
min	9.729000e+05	1.054900e+06	3.136300e+06	1.802000e+05
25%	2.257900e+06	2.954850e+06	1.197440e+07	1.430500e+06
50%	3.215900e+06	3.960900e+06	1.817230e+07	2.155200e+06
75%	5.204550e+06	6.122550e+06	2.613805e+07	3.928200e+06
max	1.967843e+08	3.722226e+08	1.123003e+09	1.222342e+09

Table 1: Describe

El campo es bastante complejo, a priori nuestro dataset depende del tiempo, en general los datos financieros son procesos estocásticos ya que tenemos la variable del tiempo, al obviar el factor del tiempo no estamos teniendo en cuenta algo muy importante, también habría que ver que definimos con un pico u outlier, una subida muy alta de la cotización de los retornos? Puede que este tipo de cosas sea cíclicas y normales respecto a la serie del tiempo. Para facilitar el tema elegido simplemente aplicaremos los algoritmos sobre un atributo o sobre ambos para facilitar la comparación y visualización y comentar los resultados.

En general nuestra serie temporal de volumen tiene picos significativos en el volumen de operaciones en nuestros ETF elegidos. Algunos eventos notables incluyen la caída repentina del Tesoro de octubre de 2014, el aumento de la volatilidad en agosto de 2015, así como la elección de Donald Trump a fines de 2016. Nótese la relativa calma a principios de 2017 ...

Si bien estos eventos son obvios a simple vista (mucho después del hecho), lo que sería útil es la capacidad de clasificar automáticamente los eventos basándose únicamente en los datos del volumen de operaciones, es decir, sin el uso de un ser

humano. Este es el objetivo del aprendizaje automático y, afortunadamente, este es exactamente el tipo de caso de uso para utilizar nuestros algoritmo.

Inicialmente creo que es necesario trabajar con un ejemplo simple de método de detección de anomalías con una variable únicamente en el que detectamos valores atípicos a partir de una distribución de valores en un solo espacio de características. Por ello, utilizaremos únicamente los datos del atributo de **Gamestop(GME)** y vamos a encontrar patrones en volumen de trading y closing price por separado que no se ajustan al comportamiento esperado. Es decir, detectar valores atípicos para una variable a la vez.

Mostramos una pequeña visualización inicial de nuestro conjunto de datos.

Skewness: 10.093959

Kurtosis: 122.632442

En las imagenes [4] y [5] vemos que está lejos de una distribución normal, y tiene una cola larga y delgada positiva, la masa de la distribución se concentra a la izquierda de la figura. Las distribuciones superan con creces las colas de la distribución normal. Hay una región donde los datos tienen baja probabilidad de aparecer y está en el lado derecho de la distribución.

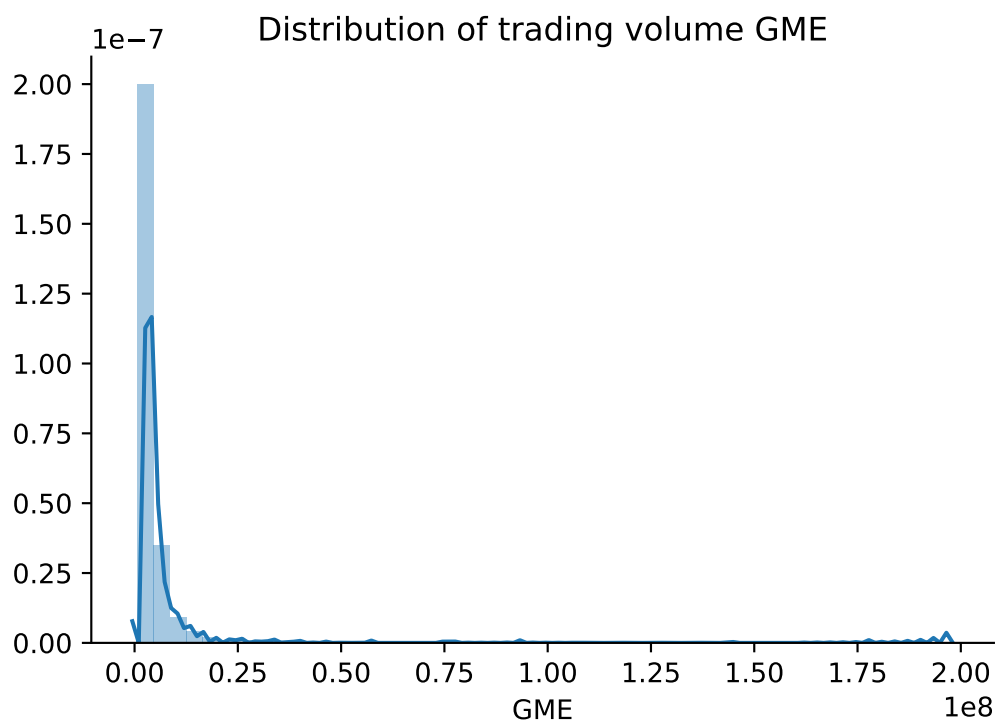


Figure 4: Distribución trading volume GME

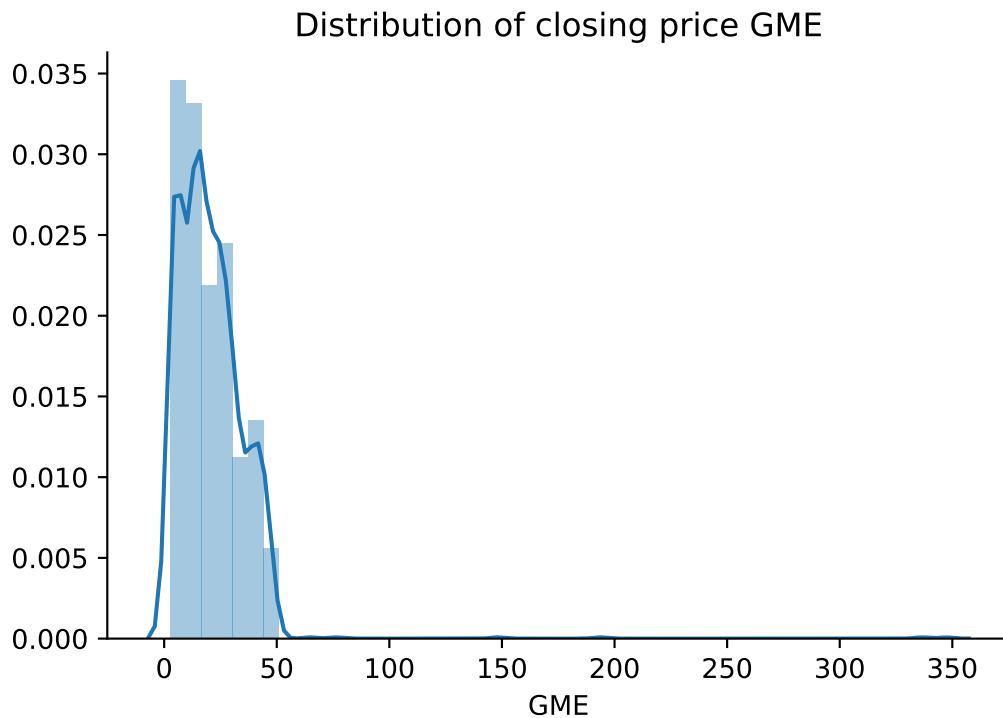


Figure 5: Distribución closing price GME

Como hemos visto antes, Isolation Forest es un algoritmo para detectar valores atípicos que devuelve la puntuación de anomalía de cada muestra utilizando el algoritmo IsolationForest que se basa en el hecho de que las anomalías son puntos de datos que son pocos y diferentes. Isolation Forest es un modelo basado en árboles. En estos árboles, las particiones se crean seleccionando primero al azar una característica y luego seleccionando un valor de división aleatoria entre el valor mínimo y máximo de la característica seleccionada. Vamos a utilizar este algoritmo que funciona para Univariate Anomaly Detection en el caso de GME.

El siguiente proceso muestra cómo se comporta IsolationForest en el caso de GME, IsolationForest entrenado usando los datos de GME. Almacenamos el valor en una matriz NumPy para usarla en nuestro modelo más adelante. Calcula la puntuación de anomalía para cada observación. La puntuación de anomalía de una muestra de entrada se calcula como la puntuación media de anomalía de los árboles en el bosque. Clasificamos cada observación como un valor atípico o no atípico. Y finalmente usamos visualización para resaltar las regiones donde caen los valores atípicos.

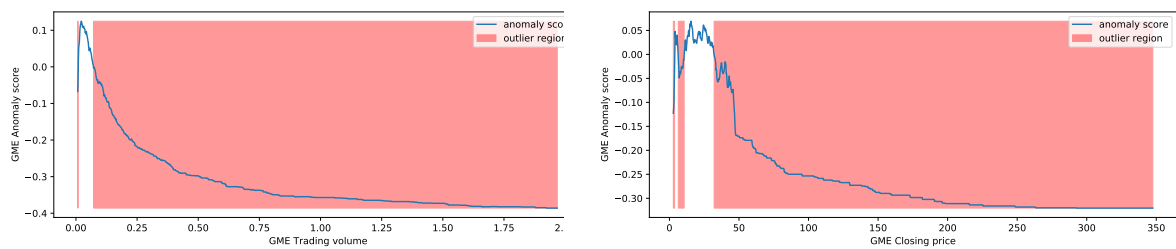


Figure 6: Isolation Forest GME

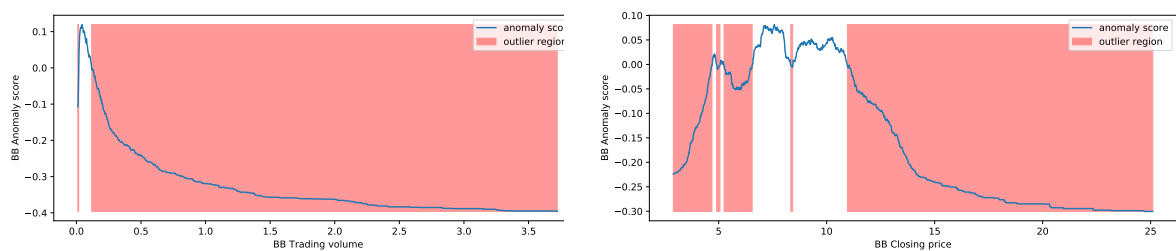


Figure 7: Isolation Forest BB

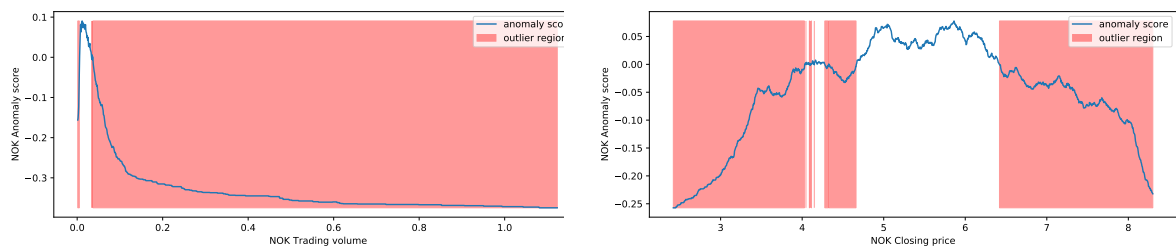


Figure 8: Isolation Forest NOK

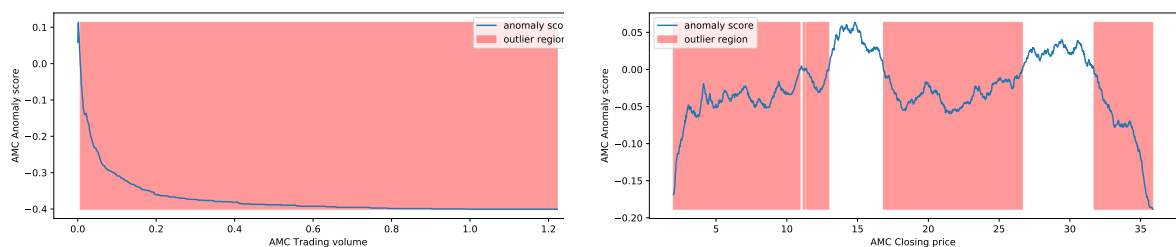


Figure 9: Isolation Forest AMC

En las imagenes de la izquierda tenemos la visualización del algoritmo con respecto el trading volume, y a la derecha respecto al closing price. Una puntuación cercana

a 1 indica anomalías. Una puntuación mucho menor que 0,5 indica observaciones normales. Si todas las puntuaciones se acercan a 0,5, entonces la muestra completa no parece tener anomalías claramente diferenciadas. Por ejemplo, en la figura de la izquierda de [6], valores por encima de 0,15 definitivamente se considera un outlier. Lo cual concuerda visualmente con la gráfica [2]. Al contrario que la imagen de la derecha donde comparado con la gráfica de la izquierda, los resultados y la visualización anteriores, parece que los beneficios que estén fuera del intervalo blanco se considerarían un valor atípico, lo cuál visualmente tiene sentido. De igual manera razonaríamos para los demás casos. Destacamos [9] donde al ser más estable la gráfica, es normal que cualquier valor que sobresalga se considere un outlier.

Planteamiento de futuro

Referencias

- [For] *How To Apply Anomaly Detection And Reap These Three Benefits*. visited on 27-01-2021. URL: <https://www.forbes.com/sites/forbesagencycouncil/2020/02/03/how-to-apply-anomaly-detection-and-reap-these-three-benefits/#5a289a5114bf>.
- [Ter] *La noche de los 40 terremotos en Granada*. URL: <https://elpais.com/espana/2021-01-27/la-noche-de-los-40-terremotos-en-granada-ibamos-buscando-espacios-sin-arboles-ni-edificios.html>.
- [Pan] *Pandas Datareader*. URL: <https://pandas-datareader.readthedocs.io/en/latest/>.
- [Pyo] *PyOD*. URL: <https://github.com/yzhao062/pyod/>.
- [Red] *Reddit group blew up GameStop stock*. URL: <https://edition.cnn.com/2021/01/27/investing/gamestop-reddit-stock/index.html>.
- [Cal] *Reddit user*. URL: https://www.reddit.com/r/wallstreetbets/comments/15nphz/gme_yolo_update_jan_26_2021/.
- [SkI] *Scikit-Learn Outlier Detection*. URL: https://scikit-learn.org/stable/modules/outlier_detection.html.