



# UCEMA

## TRADING ALGORÍTMICO Y ANÁLISIS DE SENTIMIENTO

*Trabajo Final Finanzas Quant*

# **Backtesting**

*El backtesting es la principal diferencia entre el trading discrecional y el trading cuantitativo. Es el estudio de los hipotéticos trades de una estrategia de trading que fue definida con anterioridad, con la finalidad de saber los resultados que hubiera tenido en el mercado si se hubiese aplicado, para evaluar este comportamiento se utilizan diferentes ratios y métricas de rendimiento.*

## **Bots swing -Trading del tipo trend-following**

Es una estrategia comercial según la cual uno debe comprar un activo cuando su tendencia de precios sube, se intenta aprovechar la mayor parte de dicha tendencia y se vende por un determinado take-profit, stop loss o bien una nueva señal de salida que indique que se ha perdido la tendencia.

El objetivo es capturar las tendencias del precio a corto plazo.

### **Los inputs elegidos**

Los Activos que se usarán son los siguientes: Acciones que componen al SP500 & BTC. Pertenecientes al Mercado Bursátil Americano y Cripto. Cuyo Timeframe trabajado será el diario.

### **1) Etapa de Datos**

#### **Preparación de Datos:**

Se trabajó con la fuente datos de yfinance, para continuar con el filtrado, procesado, normalizado de datos.

### **2) Etapa de investigación**

Una vez identificados los condicionantes previos, lo primero para desarrollar el sistema de trading es tener una lógica o idea inicial. La lógica del sistema está constituida por el conjunto de variables y condiciones que determinan los puntos de entrada y de salida. Constituye el núcleo de la estrategia, susceptible de ser mejorada.

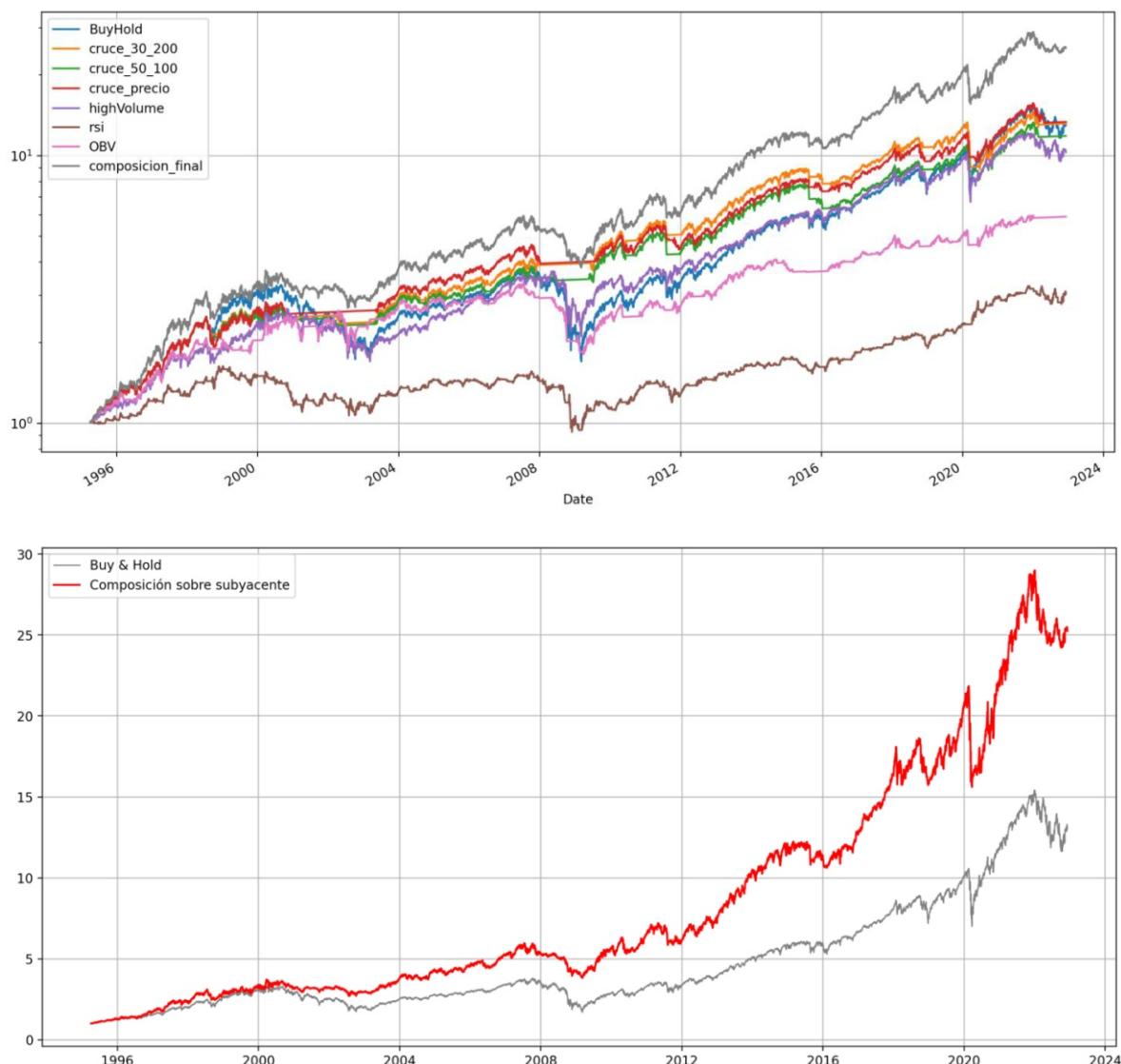
#### **Planteo de Racional:**

Primero se trabaja con la lógica operacional, parametrización y gatillos. Luego buscar indicadores que sirvan para darnos señales y por último, la construcción de indicadores/Features

#### **Luego se realiza:**

Filtro de datos, tabla de posibles trades, tiempo in/out y resultados y por último, el porcentaje de trades positivos y negativos.

1.



### **3) Etapa de análisis y métricas:**

Calcular todo tipo de ratios de la estrategia que van a ser nuestras métricas. La idea de las métricas es tener indicadores comparables de performance para después comparar estrategias y decidirse por la mejor a implementar de toda la lista que tengamos

Se realiza la tabla de resultados por trade y la comparación con el buy&hold

#### **Métricas:**

Usaremos el Enfoque Event Driven y Métricas de riesgo. Estas métricas permiten conocer el rendimiento del sistema y proporcionan un buen punto de partida para probar una estrategia de trading.

2.

## 4) Etapa de parametrización

Aquí se debe evitar el over fitting, básicamente vamos a analizar la sensibilidad de nuestro modo o racional evaluado en alguna métrica que consideremos relevante, en función de los parámetros variables del racional.

## 5) Resultados:

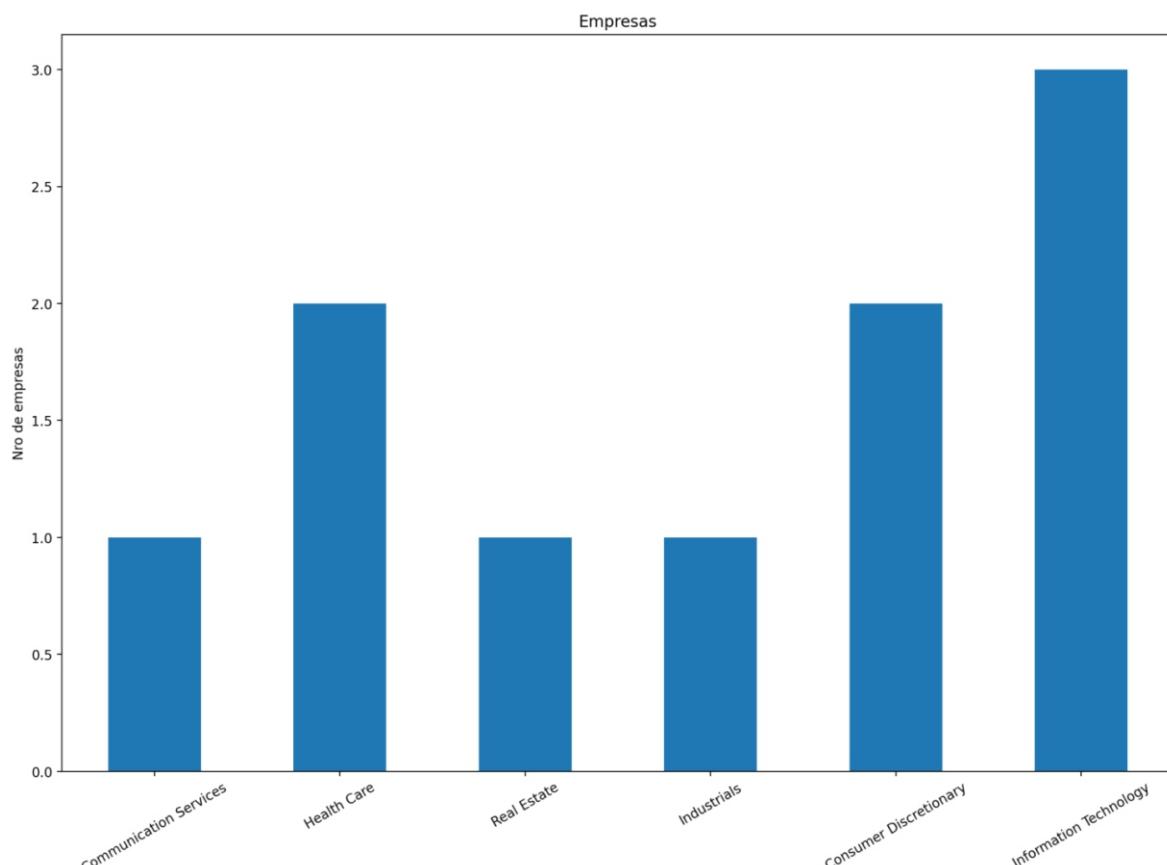
### Base de datos:

Se trabajó con WorkBench para la creación de la BD con SQL, ésta registra los trades realizados, los features y los triggers de compra/venta.

### Ploteo:

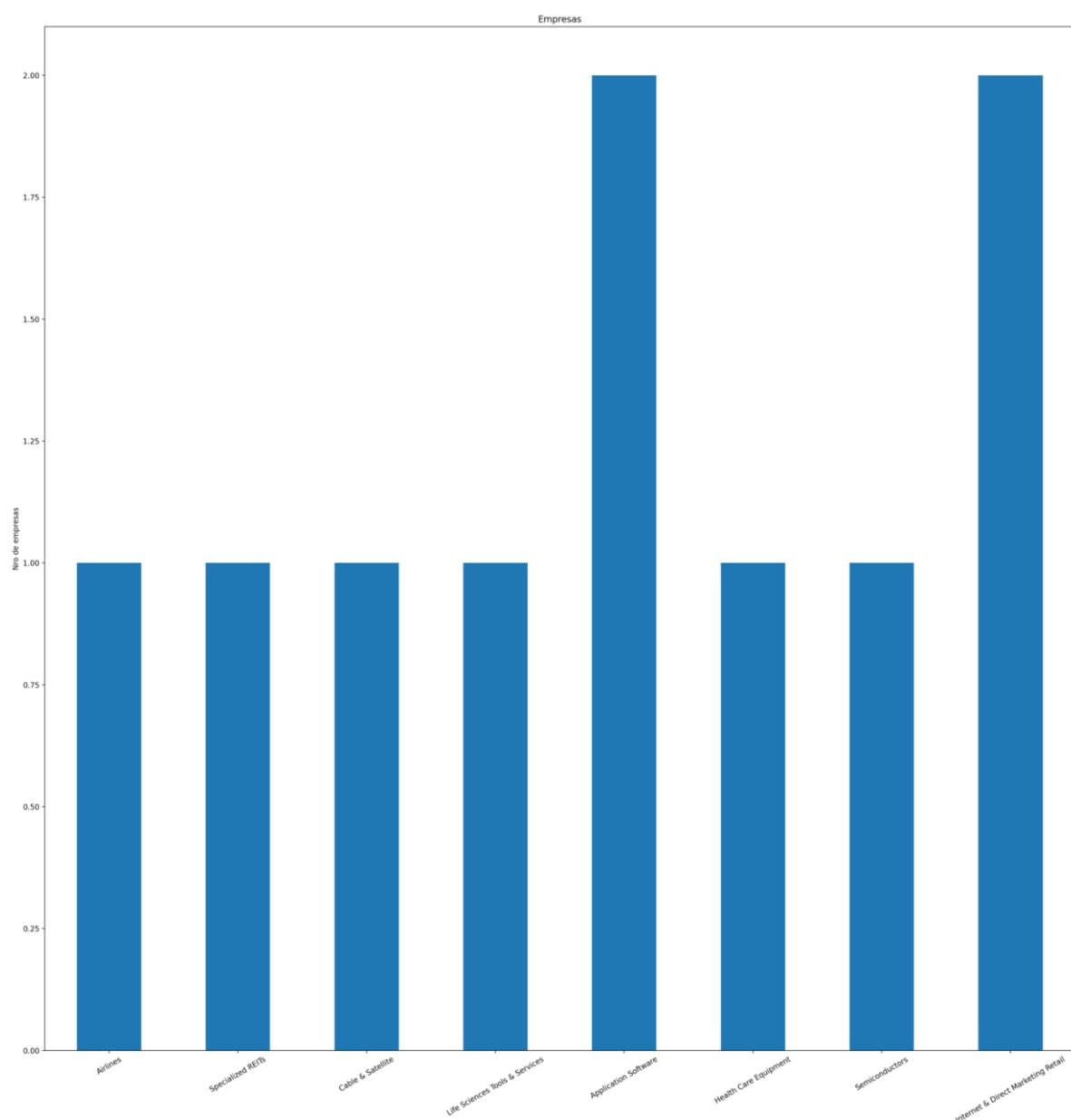
Se graficó por medio de Streamlit, un “framework” de Python de código abierto que permite de manera integrada desarrollar aplicaciones, muy usado para la ciencia de datos.

### Se realizó un filtrado de los 10 activos con mejores rendimientos:



*Gráfico de Sectores de las empresas*

3.



*Gráfico de Sub-Sectores de las empresas*

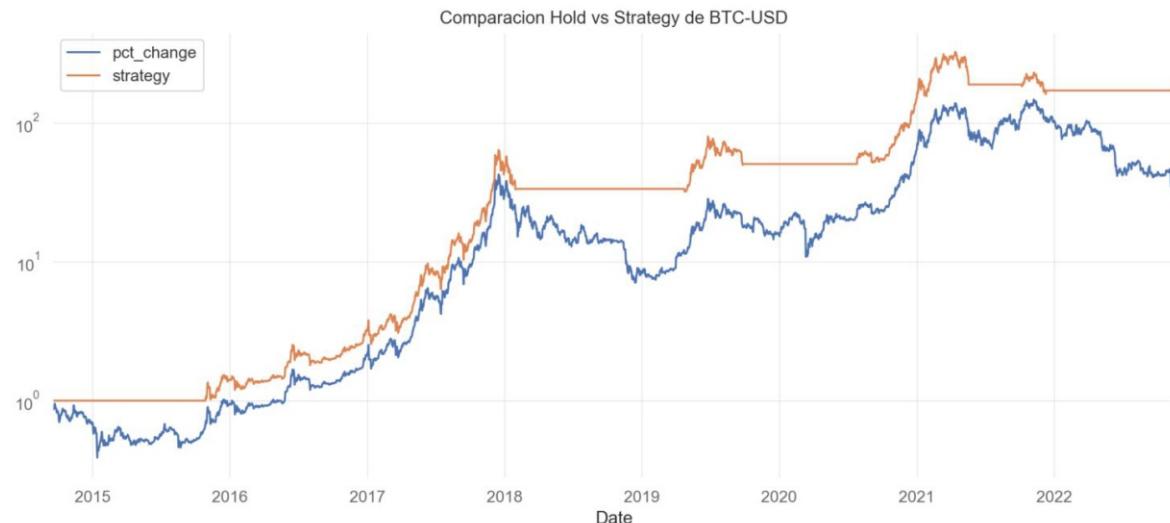
4.

## *Estrategia aplicada a la empresa AMD*



5.

## *Estrategia aplicada al Bitcoin*



6.

## *Métricas en Bitcoin de la Estrategia vs Benchmark:*

	Strategy	Benchmark
<b>Start Period</b>	<b>2014-09-17</b>	<b>2014-09-17</b>
<b>End Period</b>	<b>2022-11-20</b>	<b>2022-11-20</b>
<b>Risk-Free Rate</b>	<b>0.0%</b>	<b>0.0%</b>
<b>Time in Market</b>	<b>46.0%</b>	<b>100.0%</b>
<b>Cumulative Return</b>	<b>17,027.66%</b>	<b>3,738.43%</b>
<b>CAGR%</b>	<b>87.52%</b>	<b>56.19%</b>
<b>Sharpe</b>	<b>1.22</b>	<b>0.81</b>
<b>Prob. Sharpe Ratio</b>	<b>100.0%</b>	<b>99.73%</b>
<b>Smart Sharpe</b>	<b>1.2</b>	<b>0.8</b>
<b>Sortino</b>	<b>1.9</b>	<b>1.18</b>
<b>Smart Sortino</b>	<b>1.87</b>	<b>1.16</b>
<b>Sortino/V2</b>	<b>1.35</b>	<b>0.84</b>
<b>Smart Sortino/V2</b>	<b>1.32</b>	<b>0.82</b>
<b>Omega</b>	<b>1.41</b>	<b>1.41</b>
<b>Max Drawdown</b>	<b>-50.28%</b>	<b>-83.4%</b>
<b>Longest DD Days</b>	<b>585</b>	<b>1079</b>
<b>Volatility (ann.)</b>	<b>43.25%</b>	<b>61.14%</b>
<b>R^2</b>	<b>0.5</b>	<b>0.5</b>
<b>Information Ratio</b>	<b>0.0</b>	<b>0.0</b>
<b>Calmar</b>	<b>1.74</b>	<b>0.67</b>
<b>...</b>		
<b>Beta</b>	<b>0.5</b>	<b>-</b>
<b>Alpha</b>	<b>0.28</b>	<b>-</b>
<b>Correlation</b>	<b>70.76%</b>	<b>-</b>
<b>Treynor Ratio</b>	<b>34018.87%</b>	<b>-</b>

## 5 Worst Drawdowns

	Start	Valley	End	Days	Max Drawdown	99% Max Drawdown
1	2021-04-14	2021-12-10	2022-11-20	585	-50.276026	-46.843278
2	2017-12-17	2019-04-25	2019-06-22	552	-50.017032	-48.173489
3	2019-06-27	2019-09-26	2020-11-11	503	-37.624282	-35.154701
4	2017-09-02	2017-09-14	2017-10-12	40	-35.508102	-26.760574
5	2017-06-12	2017-07-16	2017-08-05	54	-34.761727	-32.427803

## Strategy Visualization

Cumulative Returns vs Benchmark



8.

### Cumulative Returns vs Benchmark (Log Scaled)

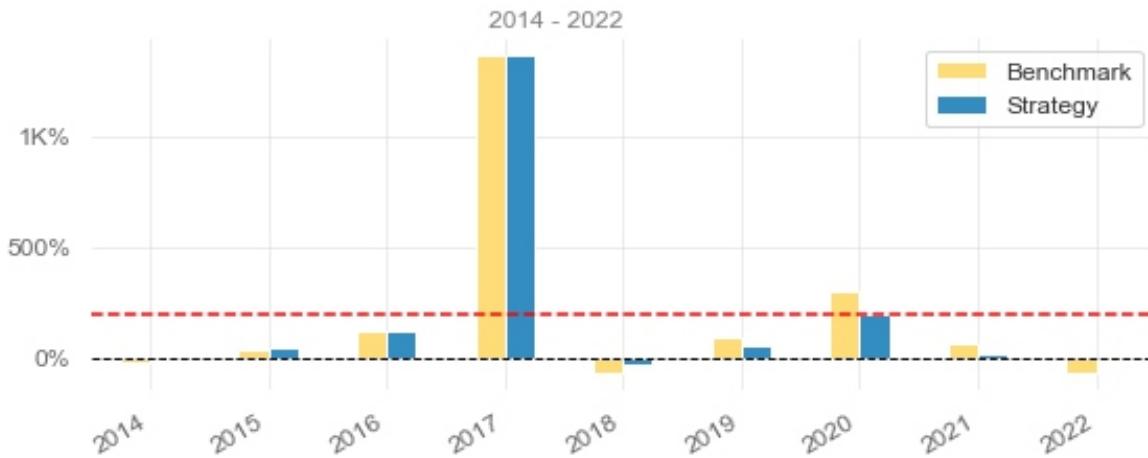


### Cumulative Returns vs Benchmark (Volatility Matched)

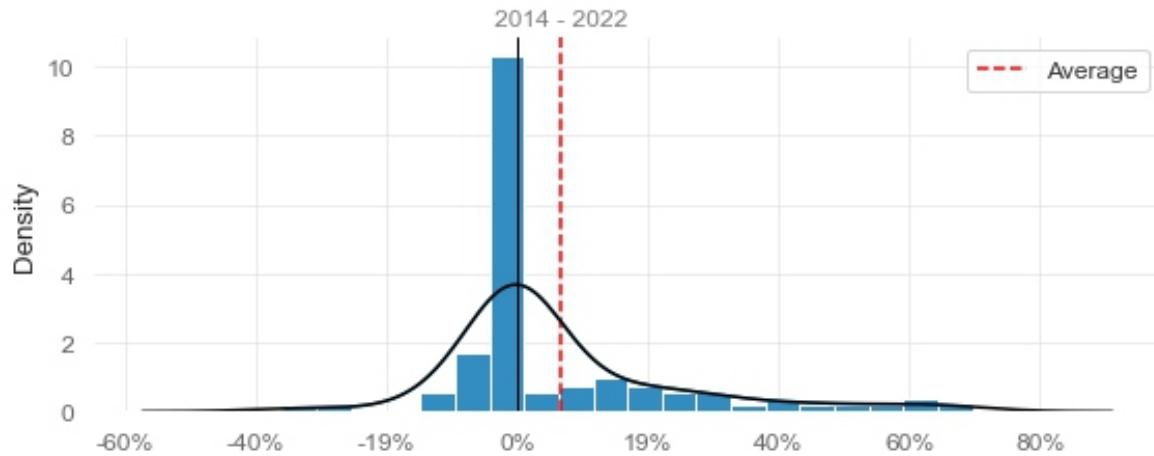


9.

### EOY Returns vs Benchmark



### Distribution of Monthly Returns



### Daily Returns



10.

### **Rolling Beta to Benchmark**



### **Rolling Volatility (6-Months)**



### **Rolling Sharpe (6-Months)**



### **Rolling Sortino (6-Months)**



## Worst 5 Drawdown Periods



## Underwater Plot



12.

### Monthly Returns (%)

2014	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2015	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.13	20.10	14.11	
2016	-14.35	18.69	-4.79	7.58	18.53	26.71	-7.23	-7.88	5.95	14.96	6.38	29.24
2017	0.69	21.60	-9.17	25.76	69.63	8.50	15.90	63.58	-7.75	49.09	58.21	38.33
2018	-27.80	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2019	0.00	0.00	0.00	-2.09	60.25	26.15	-6.76	-4.51	-14.32	0.00	0.00	0.00
2020	0.00	0.00	0.00	0.00	0.00	0.00	14.32	3.16	-7.67	27.79	42.41	47.77
2021	14.18	36.31	30.53	-1.98	-35.93	0.00	0.00	0.00	0.00	10.76	-7.03	-12.12
2022	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC

### Return Quantiles



### Análisis de Sentimiento:

El análisis de sentimiento es el proceso de determinar el tono emocional detrás de una serie de palabras. La herramienta de análisis de sentimiento es una técnica automatizada para extraer información significativa de los clientes, relacionada con sus actitudes, emociones y opiniones.

A menudo, las personas reaccionan ante cualquier evento o noticia en redes sociales, en este caso en Twitter. Cuyas publicaciones se pueden utilizar para comprender sus reacciones a los eventos o el impacto que han generado y así, con el análisis de datos, se podría crear una estrategia para la toma de decisiones.

13.

### **Cómo funciona el análisis de sentimiento**

1. Mediante el análisis del sentimiento, queremos lograr entender cuál es la intención exacta de una publicación en twitter.
2. Posteriormente queremos conocer que valoración tiene dicha frase, y para ello se le aplica la denominada polaridad, a través de la cual se clasifica el mensaje en función de la intención que tenga el autor al realizarlo, pudiendo ser este positivo, neutro o negativo.
3. Para aplicar esta polaridad y posteriormente poder obtener datos concluyentes y predecir comportamientos futuros.

*Se realizan los siguientes pasos:*

- Filtración de datos: En primer lugar, se utilizan las palabras claves para descartar contenido no deseado, y posteriormente se establecen palabras para obtener categorías según su polaridad y subjetividad. En este caso analizamos el índice del dólar y usamos la palabra clave #DXY.
- Extracción del contenido: Una vez que pasen el filtro, se elimina el contenido no deseado y se comenzará a trabajar con el contenido de calidad.
- Análisis de contenido: Este proceso lo realiza el algoritmo. Aquí el contenido útil y de calidad quedará encuadrado en la categoría que le corresponda.
- TextBlob: Es una biblioteca de Python para procesar datos textuales. Proporciona una API simple para sumergirse en tareas comunes de procesamiento de lenguaje natural (NLP), como el etiquetado de partes del discurso, la extracción de frases nominales, el análisis de sentimientos, la clasificación, la traducción y más.
- Por último, como salida del proceso: se obtienen las publicaciones clasificadas en negativas, positivas y neutras, como así también un gráfico de polaridad y subjetividad.

### **Estadísticas de Polaridad**

Media 0.038279186059115636

Max 1.0

Min -0.8749999999999998

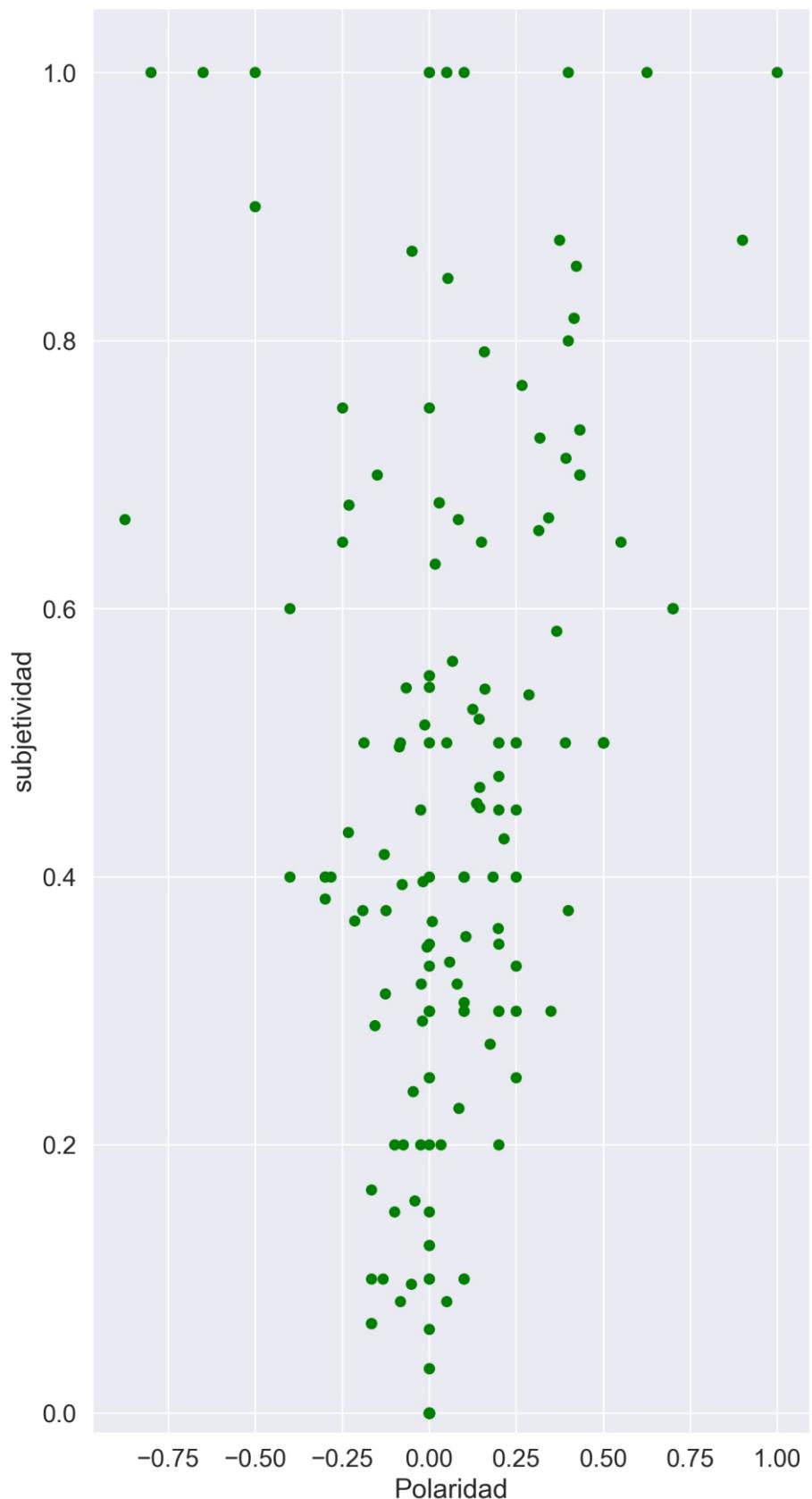
### **Estadísticas de Subjetividad**

Media 0.1882868515808657

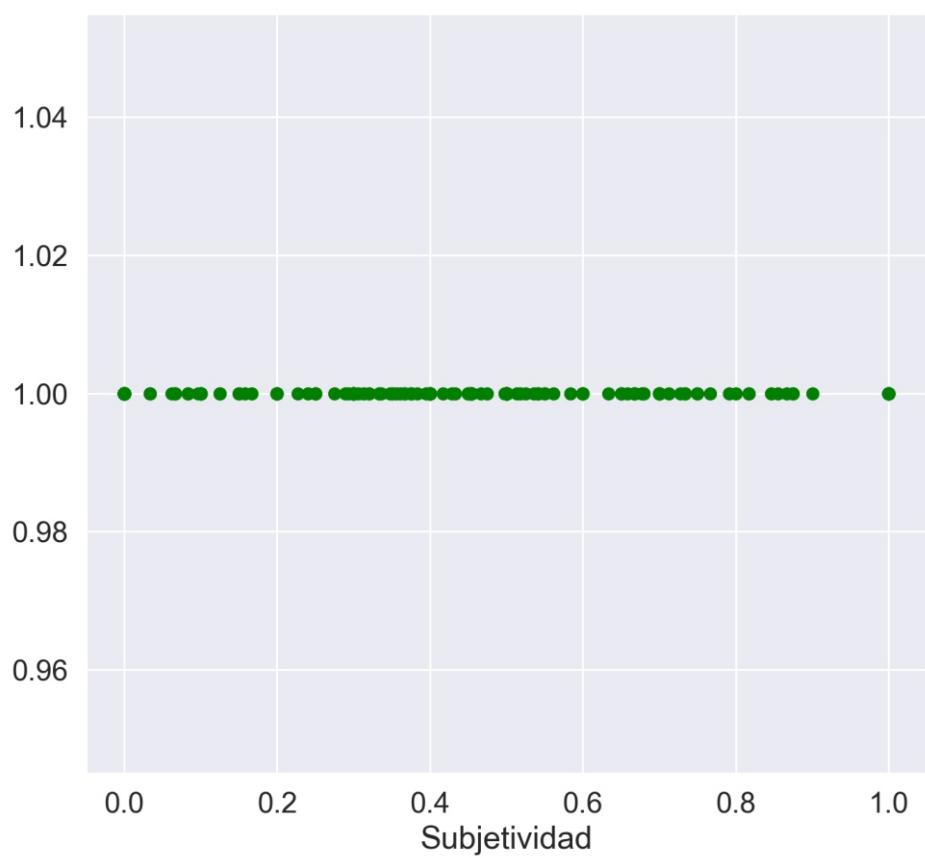
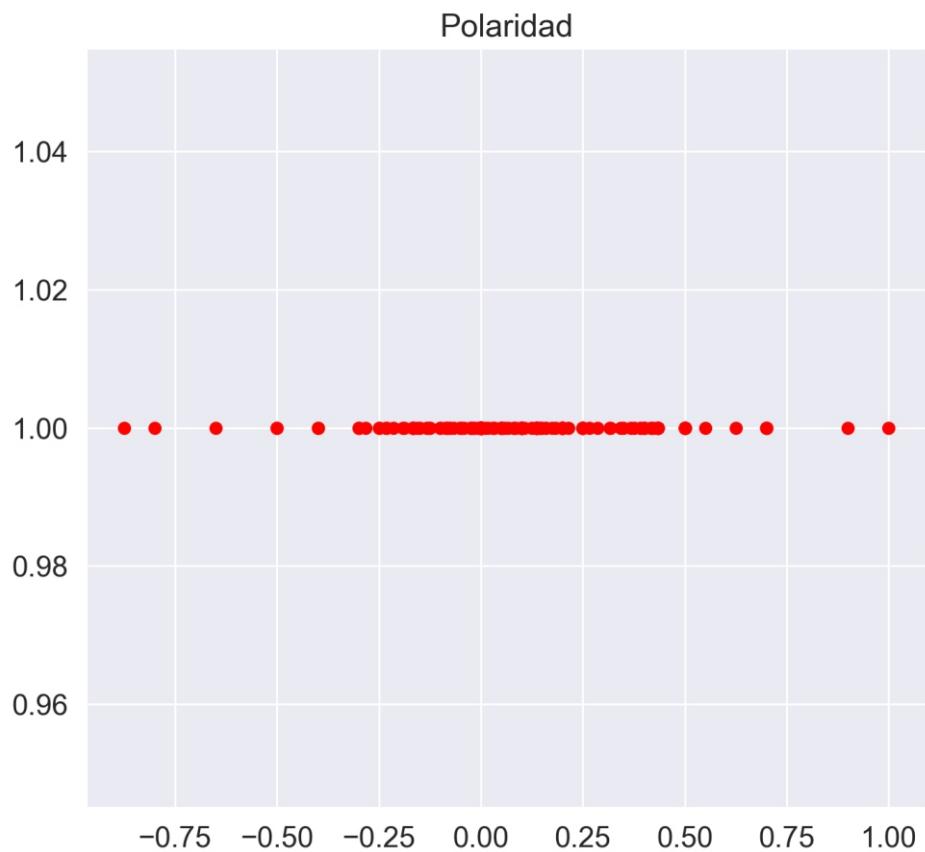
Max 1.0

Min 0.0

### Analisis de sentimiento

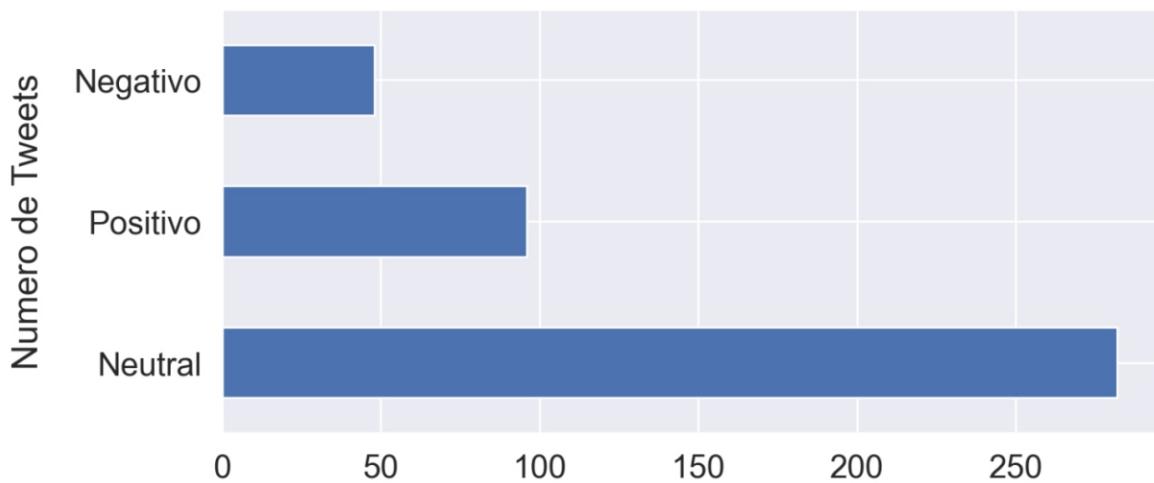


15.



16.

Analisis de sentimiento Bar plot



Neutral

