

Trabajo fin de grado

Aplicación de técnicas de análisis de series temporales a tráfico de red



Bhavuk Sikka Bajaj

Escuela Politécnica Superior
Universidad Autónoma de Madrid
C\Francisco Tomás y Valiente nº 11

UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

**Aplicación de técnicas de análisis de series
temporales a tráfico de red**

Autor: Bhavuk Sikka Bajaj

Tutor: Jorge Enrique López de Vergara Méndez

mayo 2024

Todos los derechos reservados.

Queda prohibida, salvo excepción prevista en la Ley, cualquier forma de reproducción, distribución comunicación pública y transformación de esta obra sin contar con la autorización de los titulares de la propiedad intelectual.

La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual (*arts. 270 y sgts. del Código Penal*).

DERECHOS RESERVADOS

© 2024 por UNIVERSIDAD AUTÓNOMA DE MADRID

Francisco Tomás y Valiente, nº 1

Madrid, 28049

Spain

Bhavuk Sikka Bajaj

Aplicación de técnicas de análisis de series temporales a tráfico de red

Bhavuk Sikka Bajaj

A mis padres, por su apoyo incondicional

*The goal of forecasting is not to predict the future
but to tell you what you need to know to take meaningful action in the present*

Paul Saffo

AGRADECIMIENTOS

Agradezco de todo corazón a mis padres por su inquebrantable apoyo y sacrificio a lo largo de mi carrera universitaria. Su amor y guía han sido fundamentales en este camino. También quiero expresar mi gratitud a mi hermano por su constante apoyo y motivación. A mis amigos, quienes han sido mi sostén y compañía en cada etapa, les agradezco por compartir esta travesía conmigo.

Un agradecimiento especial a mi tutor, Jorge López de Vergara, por su invaluable orientación, paciencia y sabios consejos. Su guía ha sido fundamental en el desarrollo de este trabajo de fin de grado.

RESUMEN

En un mundo cada vez más interconectado, donde Internet desempeña un papel crucial en nuestras actividades diarias, la seguridad de los sistemas informáticos y la disponibilidad constante de servicios en línea son de suma importancia. En este contexto, los ataques de denegación de servicio (DoS, por sus siglas en inglés) representan una amenaza significativa. Con el aumento generalizado del ancho de banda en hogares, lugares de trabajo y otras instalaciones, estos ataques se han vuelto más accesibles y frecuentes. Por ello, detectar y mitigar estos ataques se vuelve fundamental para mantener la funcionalidad y la integridad de los servicios en línea.

En este estudio, se exploran mecanismos simples pero efectivos para la detección de anomalías sutiles en ventanas de corta duración, con la capacidad de implementarse en tiempo real gracias a su simplicidad. Los ataques DoS causan anomalías en el tráfico al intentar saturar los recursos disponibles, por lo que se propone utilizar modelos que se ajusten al tráfico observado para evaluar si el tráfico futuro sigue un patrón esperado o difiere significativamente.

El método sugerido se basa en transformar la serie temporal de la ventana de interés en una serie estacionaria, mediante técnicas de diferenciación de series temporales. A partir de esta, se derivan parámetros que caracterizan el comportamiento del tráfico. Se exploran dos enfoques para analizar este comportamiento: ajuste de parámetros α -estables y estandarización de los datos, asumiendo que, al ser estacionaria, la media y la desviación estándar no varían con el tiempo.

La detección de anomalías se realiza comparando estos parámetros estimados con valores previamente establecidos. Si se observan desviaciones significativas, se identifica una anomalía en el tráfico.

PALABRAS CLAVE

Distribución α -estable, Tipificación, Serie Temporal, Diferenciación de Primer Grado, Diferenciación mediante Convolución, Holt-Winters, Regresión Logística, Descomposición Estacional, Ataque de Denegación de Servicio, Ciberseguridad.

ABSTRACT

In an increasingly interconnected world, where the Internet plays a crucial role in our daily activities, the security of computer systems and the constant availability of online services are of paramount importance. In this context, Denial of Service (DoS) attacks represent a significant threat. With the widespread increase in bandwidth in homes, workplaces, and other facilities, these attacks have become more accessible and frequent. Therefore, detecting and mitigating these attacks becomes essential to maintain the functionality and integrity of online services.

This study explores simple yet effective mechanisms for detecting subtle anomalies in short-duration windows, with the ability to be implemented in real-time due to their simplicity. DoS attacks cause traffic anomalies by attempting to saturate available resources, hence it is proposed to use models that fit the observed traffic to assess whether future traffic follows an expected pattern or differs significantly.

The suggested method is based on transforming the time series of the window of interest into a stationary series, using time series differencing techniques. From this, parameters characterizing traffic behavior are derived. Two approaches are explored to analyze this behavior: fitting α -stable parameters and standardizing the data, assuming that, being stationary, the mean and standard deviation do not vary over time.

Anomaly detection is performed by comparing these estimated parameters with previously established values. If significant deviations are observed, a traffic anomaly is identified.

KEYWORDS

α -stable Distribution, Standardization, Time Series, First-Order Differencing, Convolution-Based Differencing, Holt-Winters, Logistic Regression, Seasonal Decomposition, Denial of Service Attack, Cyber-security.

ÍNDICE

1 Introducción	1
1.1 Motivación	1
1.2 Objetivos	2
1.3 Fases de realización	2
1.4 Organización de la memoria	4
2 Estado del arte	5
2.1 Ataques en la red	5
2.2 Estudios anteriores de series temporales en tráfico de red	6
2.2.1 α -estables	7
2.2.2 <i>B-Splines</i>	7
2.2.3 Análisis de Datos Funcionales	8
2.2.4 Suavizado exponencial de Holt-Winters	8
2.3 Datos utilizados en los estudios	9
2.4 Conclusión	10
3 Análisis de datos	11
3.1 Herramientas de análisis	11
3.2 Descripción de los datos	12
3.2.1 Estacionariedad de los datos	13
3.2.2 Descomposición estacional	14
3.2.3 Holt-Winters	15
3.3 Simulación de ataque	17
3.4 Conclusión	18
4 Diseño del sistema	19
4.1 Estacionarización de los datos	19
4.1.1 Diferenciación de los datos	20
4.2 Esquema de funcionamiento	22
4.2.1 Ajuste α -estable	23
4.2.2 Normalización de los datos	24
4.3 Conclusión	27
5 Resultados obtenidos	29
5.1 Detección mediante α -estables	29

5.2 Detección mediante la media	31
5.3 Detección combinada	32
5.4 Métricas de evaluación	33
5.4.1 Detección sobre el conjunto de datos de prueba	36
5.5 Conclusión	37
6 Conclusiones	39
6.1 Trabajos futuros	40
Bibliografía	42
Acrónimos	43
Apéndices	45
A Enlace a código	47

LISTAS

Listado de ecuaciones

3.1	Modelo multiplicativo de la descomposición estacional	14
4.1	Diferenciación discreta de orden 1	20
4.2	<i>Kernel</i> de convolución para la diferenciación	20

Listado de figuras

1.1	Diagrama de Gantt de la realización del trabajo	3
2.1	Tráfico completo del dataset UGR16 [1]	10
3.1	Representación del <i>bitrate</i> de los datos de tráfico normal	12
3.2	Representación del <i>packerate</i> de los datos de tráfico normal	13
3.3	Descomposición estacional diaria de los datos de tráfico normal	14
3.4	Descomposición estacional semanal de los datos de tráfico normal	15
3.5	Predicción errática de los datos de tráfico normal con Holt-Winters	15
3.6	Mejor ajuste de los datos de tráfico normal con Holt-Winters, anulando el parámetro α	16
3.7	Predicción de los datos de tráfico normal con Holt-Winters, con MSE de $2,93 \cdot 10^{15}$	16
3.8	Predicción de los datos de tráfico normal con Holt-Winters, anulando el parámetro β	16
3.9	Predicción de los datos de tráfico normal con Holt-Winters, con MSE de $1,85 \cdot 10^{15}$	17
3.10	Simulación de un ataque de denegación de servicio en el <i>bitrate</i> en una ventana de 15 minutos	18
4.1	Proceso de la estacionarización de los datos	19
4.2	Diferenciación de los datos mediante diferenciación discreta y convolución	21
4.3	Diferenciación de los datos sujeto a un ataque	21
4.4	Proceso completo de detección de anomalías en el tráfico de red	22
4.5	Ajuste de los datos a una distribución α -estable	23
4.6	Ventana ejemplo de 15 minutos sobre la que se realiza el análisis	24
4.7	Ajuste de los datos a una distribución α -estable normalizando los datos	25
4.8	Normalización de los datos mediante la tipificación	26
5.1	Distribución de los valores del ajuste α -estable	30

5.2	Distribución de los parámetros <i>mean</i> y <i>std</i> tras la tipificación	32
5.3	Distribución de los parámetros δ del ajuste α -estable y <i>mean</i>	33
5.4	SVM de los parámetros δ y <i>mean</i> , utilizando kernel RBF	33
5.5	Matriz de confusión y curva ROC del SVM de los parámetros δ y <i>mean</i>	34
5.6	Matriz de confusión y curva ROC de la regresión logística con parámetros δ y <i>mean</i> ..	35
5.7	Matriz de confusión y curva ROC de la regresión logística con todos los parámetros ..	36
5.8	Clasificación de las ventanas del <i>dataset</i> sujeto a ataques, entre los días 09/06/2016 (jueves) y 10/06/2016 (viernes), utilizando regresión logística con todos los parámetros	36
5.9	Clasificación de las ventanas del <i>dataset</i> sin aplicar ataque, entre los días 09/06/2016 (jueves) y 10/06/2016 (viernes), utilizando regresión logística con todos los parámetros	37

Lista de tablas

3.1	Estacionariedad de los datos	13
4.1	Estacionariedad de la serie con diferenciación discreta	20
4.2	Estacionariedad de la serie con diferenciación mediante convolución	20
4.3	Ajuste de los datos a una distribución α -estable normalizando los datos, $\gamma = 1$ y $\delta = 0$	25
5.1	Características de la distribución del parámetro δ	30
5.2	Características de la distribución del parámetro <i>mean</i>	31
5.3	Correlación entre los valores de δ y <i>mean</i>	32
5.4	Resultados de la regresión logística con parámetros δ y <i>mean</i>	34
5.5	Resultados de la regresión logística con todos los parámetros	35

INTRODUCCIÓN

En este capítulo se expondrá la motivación detrás de la realización de este Trabajo de Fin de Grado (TFG), así como los objetivos que se pretenden alcanzar. También se describirán las fases en las que se ha llevado a cabo el trabajo y se presentará la estructura del documento.

1.1. Motivación

La predicción del tráfico es una componente esencial de la planificación, el desarrollo y la gestión de redes. En un mundo cada vez más conectado, donde la demanda de servicios y aplicaciones móviles continúa creciendo, es fundamental poder anticipar y adaptarse a los cambios en el tráfico de datos. La predicción del tráfico se refiere a la capacidad de estimar la cantidad de datos que se transmitirán a través de una red en un período de tiempo determinado. Esto es crucial para garantizar un rendimiento óptimo de la red, evitar congestiones y planificar eficientemente la asignación de recursos.

Es importante considerar también los posibles ataques en la red, como los Ataques de Denegación de Servicio, *Denial of Service* (DoS) y los Ataques de Denegación de Servicio Distribuidos, *Distributed Denial of Service* (DDoS). Estos ataques pueden afectar significativamente la capacidad de la red para operar normalmente, sobrecargando los recursos y causando interrupciones en el servicio. La predicción del tráfico puede ayudar a identificar anomalías y patrones asociados con estos ataques y permitir respuestas proactivas para mitigar su impacto.

En este trabajo se explorarán técnicas de predicción del tráfico estudiadas previamente por otros autores, como el método de Holt-Winters [2], o el ajuste a distribuciones α -estables [3]. Además, se investigarán técnicas de estacionarización mediante diferenciación, y su normalización, para proponer una metodología simple que permita detectar anomalías en el tráfico de red.

1.2. Objetivos

El objetivo de este trabajo es dar continuidad a las investigaciones realizadas por Benjamín Martín en su Trabajo de Fin de Grado (TFG) titulado “Detección de ciberataques de denegación de servicio mediante Funciones Características” [4] y en su Trabajo de Fin de Máster (TFM) titulado “Estudio de la predictibilidad del tráfico en Internet para la detección de anomalías sutiles” [3].

Para mejorar la precisión en la detección de ataques de anomalías sutiles en la red, nuestro enfoque se centra en analizar la cantidad de datos brutos circulando por la red. Este estudio propone un método simple para detectar anomalías. Se emplean dos series temporales que describen el ancho de banda en un enlace específico: una serie representa el tráfico normal y la otra simula un ataque aplicado al tráfico normal. Este trabajo busca identificar y diferenciar patrones anómalos con un procesamiento eficiente, sin la necesidad de un filtrado exhaustivo o un volumen masivo de datos.

El enfoque de este estudio consiste en detectar anomalías en la serie temporal del tráfico de red. Para lograr esto, se aplicarán modelos predictivos para series temporales, como el método de Holt-Winters. También se estudiarán técnicas de estacionarización mediante diferenciación, ajustes a distribuciones α -estables y la descomposición de la serie temporal basada en su estacionalidad.

1.3. Fases de realización

La realización de este trabajo se ha llevado a cabo en varias etapas, las cuales se describen en la figura 1.1. En primer lugar, se realizó un estudio del estado del arte en la detección de ataques en la red, en el que se revisaron trabajos previos, como el TFG y el TFM de B. Martín [3, 4], y se analizaron datasets, como el UGR16 [1] y el CIC-DDoS2019 de la Universidad de New Brunswick (UNB) [5]. Además, se estudiaron técnicas de predicción de series temporales, como el uso de B-Splines, que los utiliza I. Strelkovskaya *et al.* en su artículo “Spline-Extrapolation Method in Traffic Forecasting in 5G Networks” [6], aunque finalmente no se obtuvo conclusiones relevantes de estos estudios. Luego se revisó el trabajo de E. Cabornero en su TFG [2], que trata sobre la predicción de tráfico en redes de comunicaciones con métodos de suavizado exponencial, como Holt-Winters. Esta primera etapa de reconocimiento del estado del arte se extendió durante las primeras ocho quincenas del trabajo, es decir, durante los cuatro primeros meses.

Posteriormente, se procedió a la recopilación de datos y a su análisis, para el cual se realizaron numerosas pruebas con el fin de obtener resultados comparables con trabajos anteriores. Entre estas pruebas, analizamos los datos mediante su descomposición en tendencia, estacionalidad y residuo, y se estudiaron las propiedades de estacionariedad de los datos. Esta etapa se extendió durante cuatro meses, aproximadamente.

Una vez se tuvo una idea clara de cómo se comportaban los datos, se procedió al diseño del sistema.

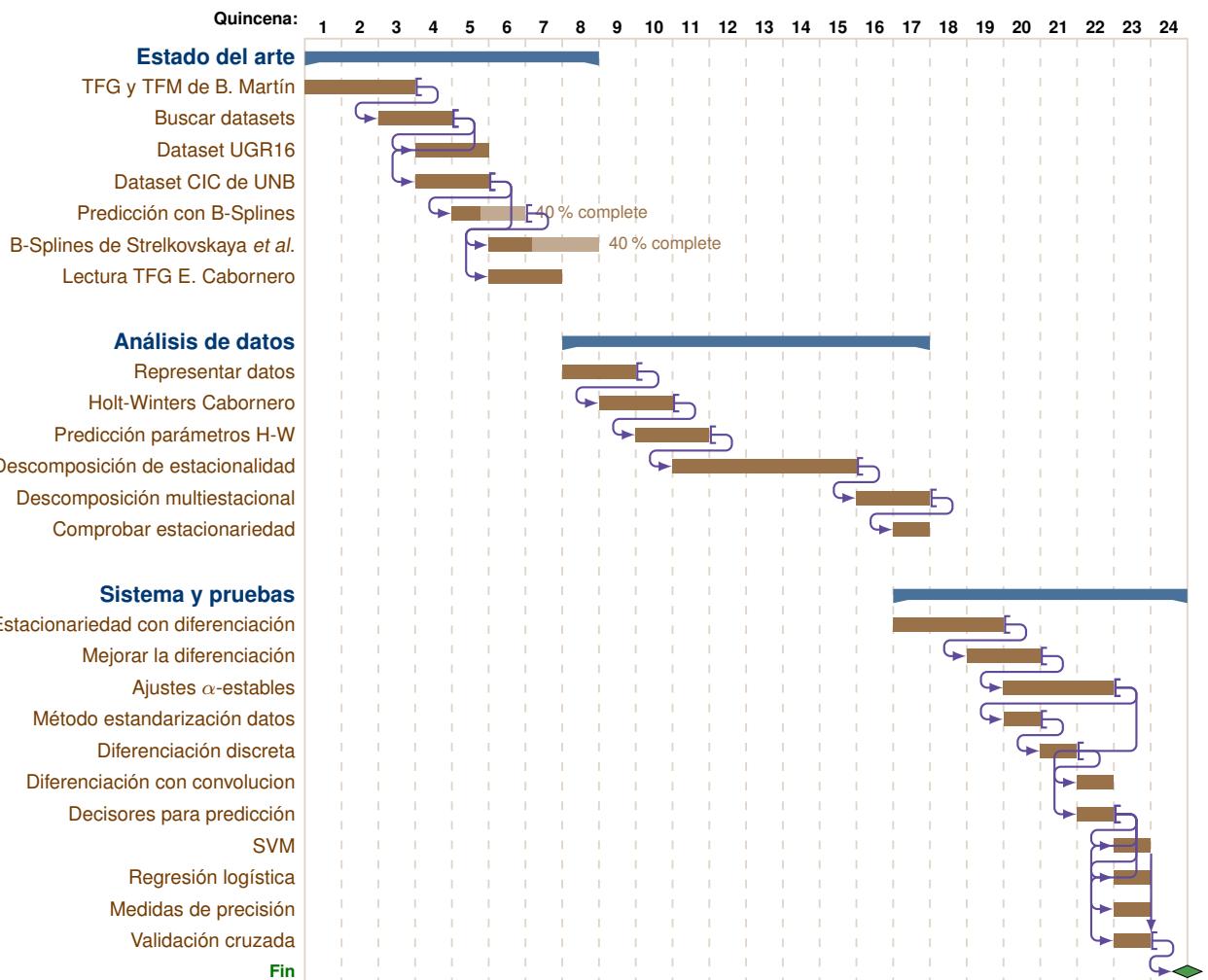


Figura 1.1: Diagrama de Gantt de la realización del trabajo.

ma de detección de ataques. En esta etapa, se comprobó la estacionariedad de los datos y se pusieron a prueba métodos de predicción basados en los ajustes α -estables que utilizó Benjamín Martín en su TFG [4]. Además, se propuso un método de estandarización de los datos basado en la media, para el cual se realizaron pruebas con diferentes métodos de diferenciación. Dicha estandarización se utilizó para la detección de anomalías en el tráfico, y se aplicaron diferentes métodos de clasificación, como Máquinas de Vectores de Soporte, *Support Vector Machine* (SVM) y regresión logística, para evaluar la precisión de la detección. Finalmente, se realizaron pruebas con los datos y se obtuvieron los resultados que se presentan en el capítulo 5. Esta etapa se extendió durante tres meses.

1.4. Organización de la memoria

El resto de la memoria se organiza de la siguiente manera:

- 1.– En el capítulo 2 se presenta el estado del arte en la detección de ataques en la red, así como los estudios anteriores de series temporales en tráfico de red.
- 2.– En el capítulo 3 se presentan las herramientas de análisis utilizadas, la descripción de los datos y la simulación de un ataque.
- 3.– En el capítulo 4 se presenta el diseño del sistema de detección de ataques, que propone un método de detección basado en la estacionariedad de los datos.
- 4.– En el capítulo 5 se presentan los resultados obtenidos con el sistema de detección de ataques.
- 5.– En el capítulo 6 se presentan las conclusiones obtenidas y los trabajos futuros propuestos para continuar con la investigación.

ESTADO DEL ARTE

La investigación sobre detección de anomalías en redes aborda dos áreas cruciales: los tipos de ataques en la red y los estudios previos que emplean series temporales para analizar el tráfico. En este contexto, es esencial comprender los principios de seguridad en Internet y la amenaza que representan los ataques DoS. Estos ataques buscan saturar los recursos de red legítimos para interrumpir servicios.

Desde una perspectiva más específica, esta investigación se enfoca en los ataques DoS de tasa reducida [7], como ejemplo de anomalía sutil que puede pasar desapercibida en métodos de detección convencionales, pero que afectan igualmente a la prestación del servicio atacado. El análisis de series temporales en el tráfico de red ha sido explorado en estudios anteriores, utilizando técnicas como distribuciones α -estables, *B-Splines*, Análisis de Datos Funcionales, *Functional Data Analysis* (FDA) y suavizado exponencial de Holt-Winters. Estas metodologías permiten modelar y predecir el comportamiento del tráfico, identificando patrones anómalos que podrían indicar actividades maliciosas.

La investigación se apoya en datos específicos, obtenidos por su calidad y relevancia en trabajos previos.

Las siguientes secciones presentan algunos de los ataques de red más comunes y se comparan algunos estudios previos en los que este trabajo se sustenta. Además, se especifica el dataset utilizado en este trabajo.

2.1. Ataques en la red

Un ataque de denegación de servicio busca incapacitar un sistema informático que proporciona servicios a los usuarios mediante el envío masivo de solicitudes que sobrecargan la capacidad de procesamiento del servidor. Como resultado, los usuarios que estén utilizando el servicio en ese momento pueden perder la conexión hasta que la situación se normalice. Estos ataques siguen siendo comunes debido a su facilidad de ejecución y el daño que causan a las entidades proveedoras de servicios. Existen diversas modalidades para llevar a cabo un ciberataque, siendo uno de los más comunes el de DoS y su variante distribuida DDoS.

Numerosos sistemas de protección contra ataques DoS utilizan la dirección IP como criterio para bloquear paquetes provenientes de un nodo identificado como potencial atacante. Esta estrategia puede no ser efectiva para detectar un ataque DDoS, ya que este tipo de ataques involucra múltiples direcciones IP. Sin embargo, en este estudio se propone una solución que independiente del método de ataque, empleando un sistema de detección basado en el análisis del comportamiento del ancho de banda durante la ocurrencia de tales ataques.

Algunos de los métodos más comunes para llevar a cabo el ataque son los siguientes [8]:

- **UDP Flood:** En este ataque, se envían grandes volúmenes de paquetes del Protocolo de Datagramas de Usuario, *User Datagram Protocol* (UDP) a un servidor o red. Como UDP es un protocolo sin conexión, el servidor no espera confirmación de recepción, lo que facilita la sobrecarga de recursos.
- **ICMP Flood (Ping Flood):** Consiste en enviar una gran cantidad de paquetes del Protocolo de Mensajes de Control de Internet, *Internet Control Message Protocol*, (ICMP) *Echo Request* (también conocido como *ping*) a un sistema, con el objetivo de agotar los recursos de procesamiento del servidor al procesar y responder a cada solicitud.
- **HTTP Flood:** Este ataque implica enviar un gran volumen de solicitudes del Protocolo de Transferencia de Hipertexto, *Hypertext Transfer Protocol* (HTTP) válidas pero maliciosas a un servidor web, con el propósito de agotar los recursos de procesamiento del servidor al atender estas solicitudes.
- **Slowloris:** En este ataque, el atacante envía múltiples solicitudes HTTP incompletas y mantiene las conexiones abiertas el mayor tiempo posible, consumiendo los recursos del servidor al mantener muchas conexiones simultáneas abiertas.

En cualquiera de estos casos, observamos un incremento sustancial en el uso del ancho de banda de la red, ya sea en número de bytes o de paquetes, lo que puede ser detectado mediante el análisis de series temporales. En este trabajo, se explorará cómo detectar esta anomalía.

2.2. Estudios anteriores de series temporales en tráfico de red

Para poder realizar una investigación de valor, es necesario presentar los estudios previos que han abordado la detección de anomalías en tráfico de red mediante el uso de series temporales. Como continuación del TFM de Benjamín Martín, esta investigación utiliza trabajos relacionados que han sido de gran interés para comprender y avanzar en este campo. Además, durante la realización de este trabajo se han encontrado otros estudios que han abordado la detección de anomalías en tráfico de red utilizando diferentes enfoques y técnicas. A continuación, se presentan algunos de estos estudios y se discuten sus contribuciones y hallazgos.

2.2.1. α -estables

La utilización de distribuciones α -estables [9] ha sido un enfoque importante en la detección de anomalías en el tráfico de red, como se discute en los trabajos de Benjamín Martín [3], Mateo Stoppa [10], y Federico Simmross [11].

F. Simmross es reconocido como pionero en el uso de distribuciones α -estables para la detección de anomalías en el tráfico de red. Propuso un método novedoso basado en un modelo α -estable de primer orden no restringido y pruebas estadísticas de hipótesis. El estudio proporciona evidencia de que la distribución marginal del tráfico real se modela adecuadamente con funciones α -estables y clasifica patrones de tráfico mediante una Prueba de Razón de Verosimilitud Generalizada, *Generalized Likelihood Ratio Test* (GLRT). Se enfoca en detectar dos tipos de anomalías: inundaciones (*floods*) y aglomeraciones repentinas (*flash-crowds*).

El trabajo de M. Stoppa también se centra en el modelado del tráfico de red utilizando distribuciones α -estables. Compara parámetros obtenidos de datos de tráfico recopilados a través de protocolos como SNMP y NetFlow para entender mejor las características estadísticas del tráfico y su capacidad para ajustarse a estas distribuciones.

B. Martín propone una técnica de detección de anomalías sutiles que combina modelos predictivos de series temporales basados en Redes Neuronales Recurrentes, *Recurrent Neural Networks* (RNN) con estadísticos. Su estudio utiliza una distribución α -estable como referencia para modelar el comportamiento usual del tráfico, de modo que la detección de anomalías se basa en identificar desviaciones significativas de este modelo esperado.

Estos estudios subrayan la importancia y utilidad de las distribuciones α -estables en la modelización y detección de anomalías en el tráfico de red. Además, proporcionan un marco teórico robusto para caracterizar el comportamiento del tráfico y permiten identificar anomalías mediante la comparación con un modelo de referencia. Sin embargo, el trabajo de M. Stoppa se limita al modelado del tráfico SNMP y NetFlow, y el enfoque de B. Martín con RNN incrementa considerablemente el coste computacional de su solución. El trabajo presente busca realizar predicciones de anomalías en la red limitando la complejidad del problema.

2.2.2. *B-Splines*

Una *B-Spline* (*Basis Spline*) [12] es una función matemática utilizada para aproximar y suavizar datos mediante una combinación lineal de funciones *spline* base locales. Cada función *spline* base es un polinomio definido en un intervalo específico y se utiliza para definir segmentos suaves de una curva. La *B-Spline* completa se forma al combinar estas funciones *spline* base de manera suave y continua, proporcionando una representación flexible de datos y curvas.

En el estudio de Strelkovskaya *et al.* [6], se aborda la predicción del tráfico autosemejante en redes móviles mediante el uso de diversas funciones *spline*, incluyendo lineales, cúbicas y *B-Splines* cúbicas. Estas funciones *spline* se emplean para predecir el tráfico autosemejante fuera de los períodos de transmisión de datos, aprovechando la capacidad de las *B-Splines* para modelar de manera flexible y suave los patrones complejos presentes en el tráfico de red.

Particularmente, Strelkovskaya *et al.* resaltan la eficacia de las *B-Splines* cúbicas en mejorar la precisión de la predicción del tráfico autosemejante. Al utilizar *B-Splines*, su estudio demuestra que es posible realizar pronósticos precisos tanto a corto como a largo plazo, lo que resulta relevante para la gestión eficiente de las redes móviles y la detección anticipada de posibles anomalías en el tráfico.

2.2.3. Análisis de Datos Funcionales

El FDA es una técnica utilizada para estudiar y analizar conjuntos de datos representados como funciones, donde cada observación representa una función suave definida en un dominio continuo, como el tiempo. Permite tratar las observaciones como muestras de funciones, facilitando la aplicación de métodos estadísticos y matemáticos para caracterizar y entender la variabilidad en estas funciones.

En el trabajo de Muelas [13], se empleó FDA para analizar patrones de comportamiento de redes a lo largo del tiempo, tratando las medidas de red como funciones continuas y aplicando técnicas específicas para caracterizar y comprender la dinámica de la actividad de red. Esto ha sido parte integral del enfoque para reducir la dimensionalidad de los datos y detectar anomalías en el tráfico de red.

2.2.4. Suavizado exponencial de Holt-Winters

El suavizado exponencial de Holt-Winters [14] es una extensión del método de suavizado exponencial básico que incorpora componentes de tendencia y estacionalidad. Este método se utiliza comúnmente para series de tiempo que muestran variaciones estacionales regulares, como datos de tráfico de red, así como datos económicos mensuales o trimestrales, entre otros.

Las principales características del suavizado exponencial de Holt-Winters incluyen:

- **Componente de Nivel (Level):** Representa el nivel base o promedio de la serie temporal.
- **Componente de Tendencia (Trend):** Captura la dirección y la tasa de cambio de la serie temporal a lo largo del tiempo.
- **Componente de Estacionalidad (Seasonality):** Modela las fluctuaciones periódicas o estacionales en la serie temporal.

El método de Holt-Winters utiliza fórmulas de suavizado exponencial para actualizar y estimar estos componentes en cada período de tiempo, permitiendo realizar pronósticos futuros basados en la

información histórica de la serie temporal.

En el trabajo de Cabornero [2], Holt-Winters se utiliza como técnica de pronóstico para modelar y predecir series temporales de tráfico de red que exhiben patrones. Esto permite analizar y anticipar patrones estacionales, e incluso tendencias en el comportamiento del tráfico.

Además, el método de Holt-Winters se puede ajustar según tres parámetros de suavizado: α para la componente de nivel, β para la componente de tendencia y γ para la componente de estacionalidad. Estos parámetros controlan la influencia de las observaciones pasadas y la velocidad de adaptación a los cambios en los componentes de la serie temporal. La selección óptima de estos parámetros puede compararse con el ajuste de parámetros en otros algoritmos de optimización, como el recocido simulado (*simulated annealing*). En este caso buscamos la mejor configuración de los parámetros α , β y γ para minimizar el error en las predicciones de la serie temporal.

Simulated Annealing

Simulated Annealing, o recocido simulado en Español, es una técnica de optimización probabilística utilizada para encontrar soluciones aproximadas a problemas de optimización combinatoria [15]. En nuestro caso, este algoritmo se ha adaptado para ajustar los parámetros del suavizado exponencial de Holt-Winters.

2.3. Datos utilizados en los estudios

Para la elección de los datos utilizados para realizar el análisis y las pruebas, se ha dado prioridad a la consideración de *datasets* relevantes para la continuación del trabajo iniciado por Benjamín Martín. En particular, se ha evaluado el *dataset* UGR16 de la Universidad de Granada (UGR) [1], utilizado en su investigación, que abarca varios meses de tráfico de red.

Sin embargo, se han considerado otros conjuntos de datos, como los proporcionados por el Instituto de Ciberseguridad de Canadá, *Canadian Institute for Cybersecurity* (CIC) de la UNB [5]. Sin embargo, estos últimos no fueron seleccionados debido a la cantidad limitada de datos que contienen, ya que se trata de tráfico recopilado durante pocas horas durante un período de dos días únicamente. Esta elección se basa en la necesidad de disponer de datos suficientes y representativos para analizar y desarrollar métodos efectivos de detección de anomalías en el tráfico de red.

Para asegurar la calidad y relevancia de los datos, se han utilizado registros extensos de tráfico de red, que abarcan períodos de tiempo significativos y contienen una variedad de patrones de comportamiento. Estos datos incluyen mediciones de flujo de red, paquetes capturados en diferentes segmentos de la red y parámetros específicos relacionados con el tráfico. Dado que la información se presenta en forma de flujos, es necesario procesarla para convertirla en datos de bits y paquetes por segundo. Es

importante destacar que Benjamín Martín ya ha realizado esta labor en su TFG [4], lo que facilita la comparación y la continuidad del trabajo.

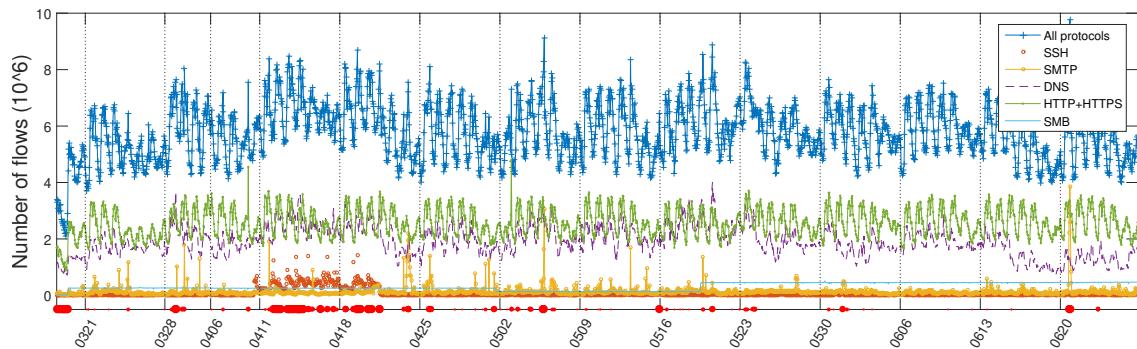


Figura 2.1: Tráfico completo del dataset UGR16 [1]

En la figura 2.1 se muestra el tráfico completo del *dataset* UGR16, que se utilizará en el desarrollo de este trabajo. Debido a la presencia de anomalías ya presentes en el *dataset*, correspondiente a tráfico no usual, se consideran ventanas de tiempo reducidas para analizar, de forma que no contengan anomalías conocidas. En particular, los puntos de color rojo muestran picos de tráfico inusual. Se ha decidido tomar esta medida para evitar que las anomalías presentes en el *dataset* afecten el análisis y la detección de anomalías en el tráfico de red.

De esta forma, para la elaboración del estudio se utilizarán dos series temporales:

- **Tráfico normal:** Representa el tráfico normal sin ataques. Los datos en esta serie incluyen la tasa binaria o *bitrate* y la tasa de paquetes o *packet rate* del tráfico de red, medidos en bits y paquetes por segundo, respectivamente. Cada punto en la serie corresponde a una medición por segundo.
- **Tráfico de ataque:** Corresponde al tráfico de ataque, generado a partir del tráfico normal mediante la inyección de tráfico malicioso. Al igual que el tráfico normal, incluye mediciones de *bitrate* y *packet rate*, con una frecuencia de medición de un punto por segundo.

2.4. Conclusión

En resumen, el estado del arte abordado en este estudio ofrece una visión detallada de los enfoques y técnicas empleadas en la detección de anomalías en el tráfico de red mediante el análisis de series temporales. Se han revisado aspectos fundamentales como los tipos de ataques en la red, los métodos de modelado y análisis de series temporales, incluyendo técnicas como el ajuste a las distribuciones α -estables y el suavizado exponencial de Holt-Winters. También se destaca la importancia de utilizar datasets variados y representativos para validar los métodos propuestos.

Las metodologías introducidas en este capítulo se aplicarán y evaluarán en el siguiente capítulo utilizando el *dataset* UGR16 [1].

ANÁLISIS DE DATOS

En este capítulo exploraremos el análisis detallado de datos de tráfico de red, abordando desde la descripción inicial de los datos hasta la simulación de un ataque de denegación de servicio. Comenzaremos con un análisis de series temporales del *bitrate*, seguido por la evaluación de la estacionariedad y la descomposición estacional de estos datos. También mostraremos cómo es la predicción de tráfico con Holt-Winters. Posteriormente, nos centraremos en simular dos escenarios de ataques que nos serán de utilidad para evaluar la detección de anomalías en el tráfico de red.

3.1. Herramientas de análisis

Las herramientas predominantes en el estudio fueron MATLAB y Python. A continuación, se describen las tareas más importantes realizadas con cada una de ellas:

- **MATLAB:** Se empleó para analizar los estudios de Stresklovaya *et al.*, aunque no se obtuvieron resultados significativos. Además, MATLAB se utilizó para realizar ajustes de datos a distribuciones α -estables, debido a la complejidad del algoritmo implementado en Python, que requería un tiempo considerable para completar los ajustes. En MATLAB, un ajuste de 898 puntos tardaba 0,04 segundos de media, mientras que en Python un ajuste de 10 puntos tardaba 11 segundos de media. No se pudo obtener una estimación del tiempo precisa para un ajuste con 898 puntos en Python, puesto que el tiempo superaba 5 minutos de ejecución.
- **Python:** Fue la principal herramienta utilizada para realizar diversas pruebas y análisis en el estudio. Se utilizaron librerías como `numpy`, `pandas`, `statsmodels` y `sklearn` para el tratamiento de series temporales y otras utilidades. Las pruebas realizadas en Python incluyeron:
 - Tests de estacionariedad: Dickey-Fuller Aumentado, *Augmented Dickey-fuller* (ADF) [16], Kwiatkowski-Phillips-Schmidt-Shin (KPSS) [17] y Kolmogorov-Smirnov (K-S) [18].
 - Diferenciación de series que se abordará más adelante en el estudio.
 - Uso de ondículas o *wavelets* para el análisis de series temporales.
 - Descomposición de series utilizando la función `seasonal_decompose`.

3.2. Descripción de los datos

En esta sección, presentamos los datos utilizados en nuestro estudio, que abarcan un periodo de dos semanas de tráfico de red sin anomalías. Estos datos están representados como series temporales de *bitrate* en la figura 3.1 y *packet rate* en la figura 3.2. Cada punto en las series temporales corresponde a una medición por segundo. Los datos se presentan en forma de flujos de red, por lo que es necesario procesarlos para convertirlos en datos de *bitrate* y *packet rate*. Sin embargo, para una representación más clara, se han suavizado los datos utilizando un promedio móvil de un minuto.

En la figura 3.1, se observa un comportamiento periódico, mostrando un patrón de tráfico de red que se repite cada 24 horas, aunque con menor tráfico durante los fines de semana en comparación con los días laborables. Este patrón es común en los datos de tráfico de red, reflejando el comportamiento de los usuarios de la red en un entorno laboral. Además, se observa una doble joroba en los datos de cada día, correspondiente a los horarios de mañana y tarde. El mínimo local que se observa entre las dos jorobas se corresponde con las 2 de la tarde, cuando los usuarios salen a comer.

En la figura 3.2, se observa un comportamiento similar, pero con picos ocasionales muy pronunciados. Estos picos indican una cantidad elevada de paquetes transmitidos en un segundo, sin embargo, en los mismos periodos de tiempo, el *bitrate* no aumenta de manera significativa. Esto puede deberse a la transmisión de paquetes pequeños, lo que resulta en un aumento del *packet rate* sin un aumento significativo del *bitrate*. Debido a esta variabilidad en el *packet rate*, el análisis de este trabajo se centrará en el *bitrate*.

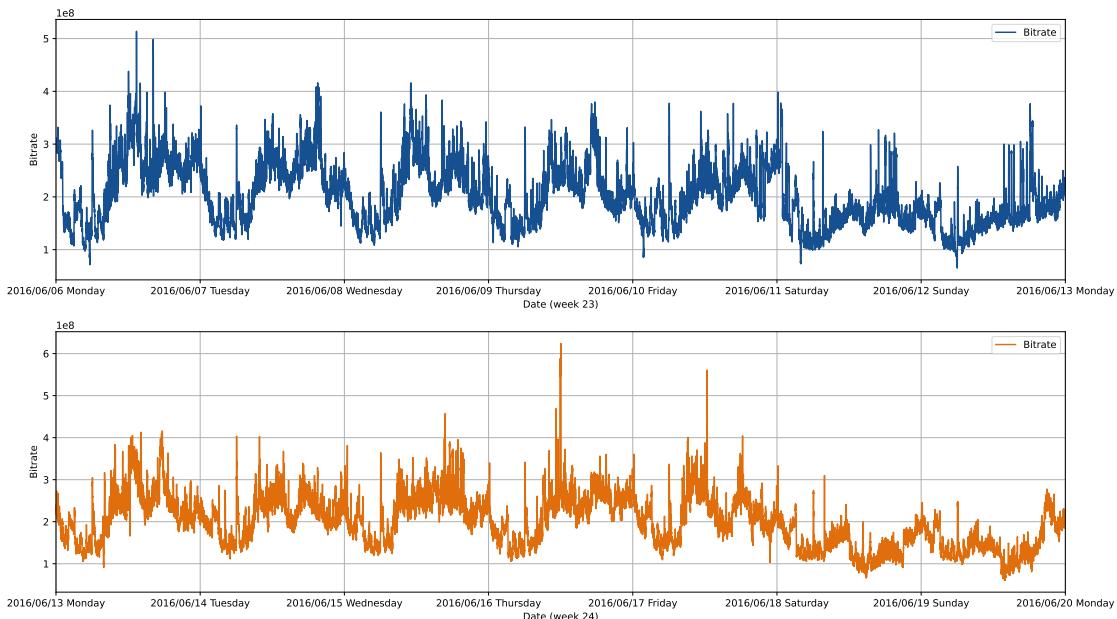


Figura 3.1: Representación del *bitrate* de los datos de tráfico normal

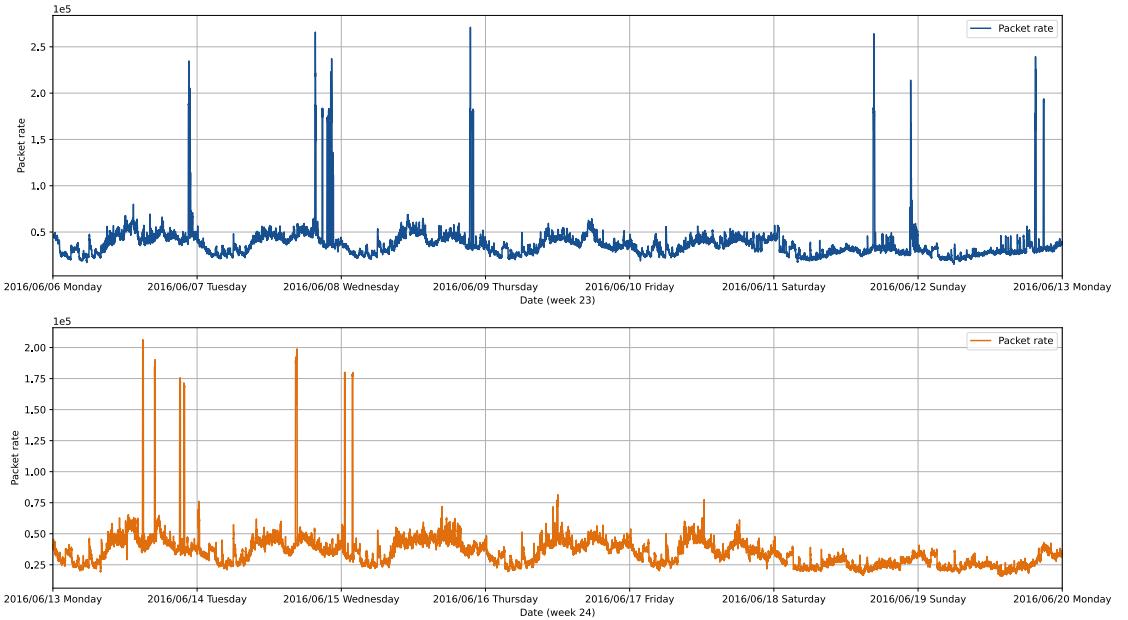


Figura 3.2: Representación del packet rate de los datos de tráfico normal

3.2.1. Estacionariedad de los datos

Es importante destacar la estacionariedad de los datos, especialmente en comparación con el trabajo de Benjamín Martín [3]. En él se realizaron pruebas de estacionariedad utilizando los tests ADF y KPSS. La hipótesis nula del test ADF es la presencia de raíces unitarias (siendo la hipótesis alternativa que la serie temporal es estacionaria), mientras que la hipótesis nula del test KPSS es que los datos son estacionarios. Si el p-valor es menor que 0,05, entonces se rechaza la hipótesis nula. Esto implica que para que los datos sean estacionarios, el p-valor debe ser mayor que 0,05 en el test ADF y menor que 0,05 en el test KPSS.

Test	P-valor medio	Porcentaje de ventanas no estacionarias
ADF	0,0595	0,1804
KPSS	0,0311	0,7665

Tabla 3.1: Estacionariedad de los datos

Se han computado ambos tests para todas las ventanas de 15 minutos en los datos y se presentan los resultados de estos tests en la tabla 3.1, que muestra que no podemos suponer estacionariedad de los datos en todas las ventanas. Por tanto, es necesario un estudio de la estacionarización de los datos antes de aplicar las técnicas de detección de anomalías propuestas por Benjamín.

3.2.2. Descomposición estacional

Para analizar estos datos, primero hemos aplicado la técnica de descomposición estacional utilizando la función `seasonal_decompose` de Python. Esta técnica nos permite visualizar y comprender la estructura estacional y tendencial de las series temporales. Se ha utilizado el modelo multiplicativo de la descomposición estacional, que se define como:

$$y(t) = T(t) \times S(t) \times R(t) \quad (3.1)$$

donde $y(t)$ es la serie temporal, $T(t)$ es la tendencia, $S(t)$ es la estacionalidad y $R(t)$ es el residuo. La tendencia representa la variación a largo plazo de los datos, la estacionalidad representa la variación periódica, que hemos fijado a 24 horas, y el residuo representa la variación aleatoria.

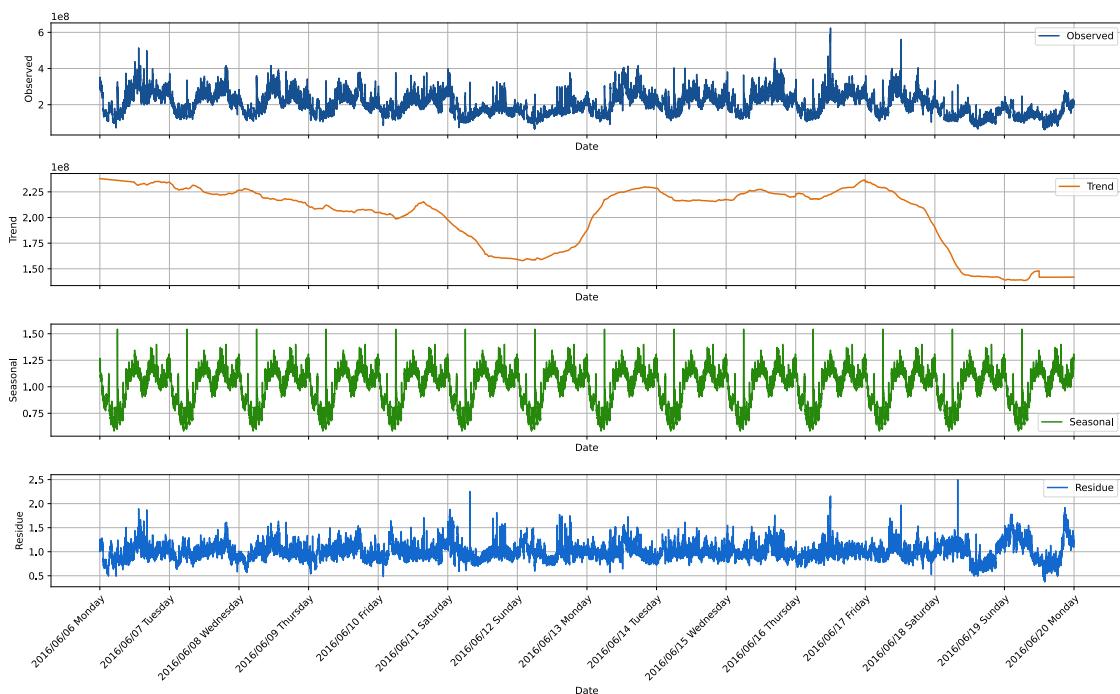


Figura 3.3: Descomposición estacional diaria de los datos de tráfico normal

En la figura 3.3, se muestra la descomposición estacional de la serie temporal del *bitrate*. Los componentes de la figura son los datos reales, la tendencia, la estacionalidad y el residuo. Para esta descomposición, se ha utilizado un periodo de 24 horas, lo que permite visualizar la variación diaria de los datos. En este caso, se observa un patrón semanal en el residuo, lo que indica que esta descomposición estacional no es perfecta; observamos asimismo que el residuo fluctúa entre 0,5 y 2,5. Por ello se ha realizado una descomposición estacional semanal.

En la figura 3.4, se muestra la descomposición estacional de la serie temporal con un periodo de 7 días. En este caso, se observa una estacionalidad semanal clara, con un patrón de tráfico que se repite cada semana. Además, se observa que el residuo es más uniforme, variando entre 0,5 y 1,5, lo

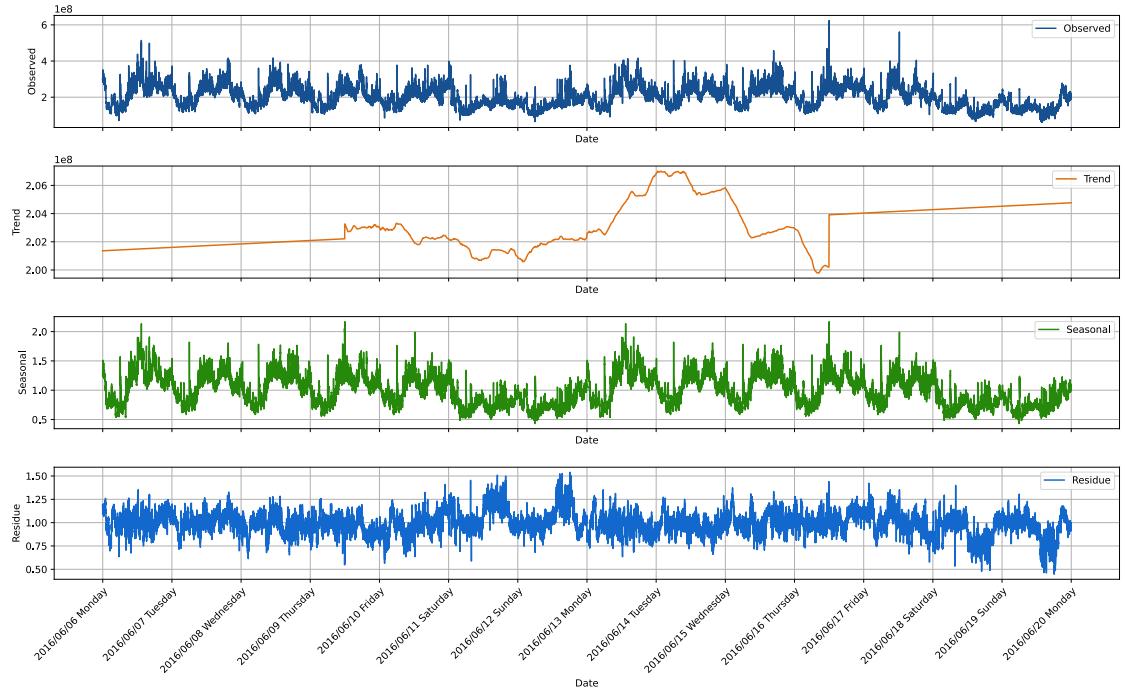


Figura 3.4: Descomposición estacional semanal de los datos de tráfico normal

que indica que la descomposición estacional es más precisa.

3.2.3. Holt-Winters

Tras haber comprobado la estacionalidad de los datos mediante `seasonal_decompose` en el apartado anterior, se observó que los datos de tráfico normal presentan una estacionalidad semanal, e incluso diaria, con una reducción de tráfico durante los fines de semana.

Por tanto, se ha aplicado el método de Holt-Winters para modelar y predecir los datos de tráfico normal. Se han realizado las pruebas teniendo 9 días de datos de entrenamiento y 1 día de datos de test. También se han calculado los errores de predicción con el Error Cuadrático Medio, *Mean Squared Error* (MSE). Las figuras se han suavizado con un promedio móvil de 10 minutos para una mejor visualización.

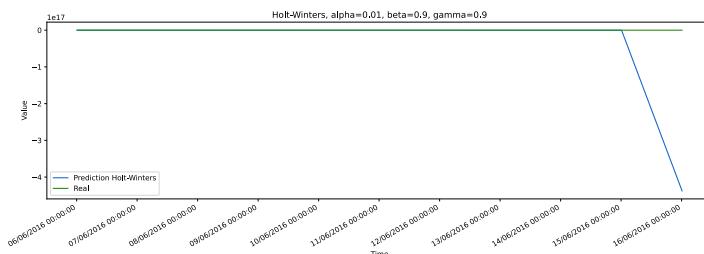


Figura 3.5: Predicción errática de los datos de tráfico normal con Holt-Winters

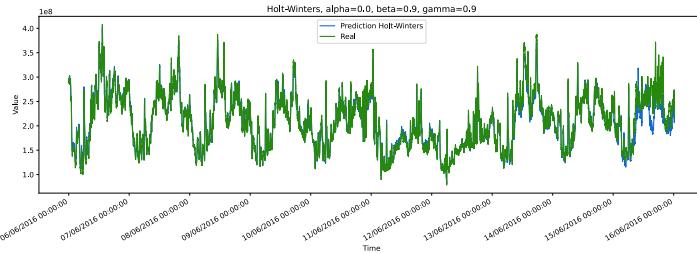


Figura 3.6: Mejor ajuste de los datos de tráfico normal con Holt-Winters, anulando el parámetro α

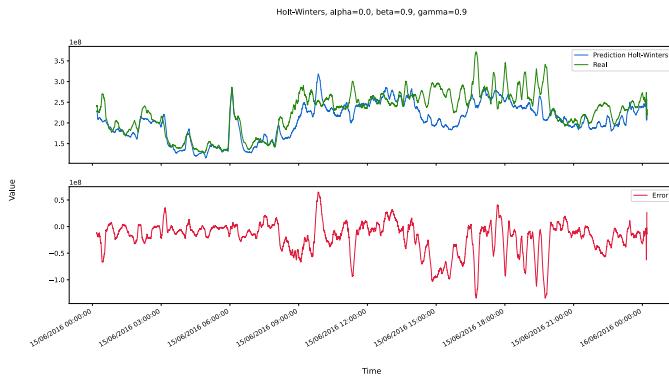


Figura 3.7: Predicción de los datos de tráfico normal con Holt-Winters, con MSE de $2,93 \cdot 10^{15}$

Se realizaron pruebas con diferentes configuraciones de parámetros α , β y γ . Se observó que para valores altos de β combinados con valores no nulos de α , las predicciones divergen considerablemente, mostrando un comportamiento errático, como se observa en la figura 3.5. Al anular el parámetro α , las predicciones se ajustan mejor a los datos de entrenamiento, como se ve en la figura 3.6, y su versión ampliada 3.7. Sin embargo, el error de predicción sigue siendo elevado, con un MSE de $2,93 \cdot 10^{15}$ en la zona de predicción.

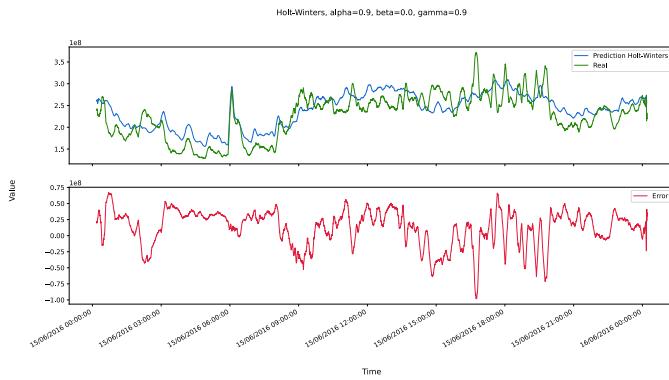


Figura 3.8: Predicción de los datos de tráfico normal con Holt-Winters, anulando el parámetro β , con MSE de $1,99 \cdot 10^{15}$

Un valor bajo de α implica que el nivel estimado no se adapta directamente a los datos observados, lo que puede resultar en mayores errores de predicción dentro de la región donde se tienen datos

disponibles. Sin embargo, se observó que las predicciones se ajustan mejor a los datos de test cuando se toma valores de α no nulos junto con valores de β nulos, como se muestra en la figura 3.8. En este caso, el MSE se reduce a $1,99 \cdot 10^{15}$.

Altos valores de γ resultan en que la predicción se adhiera al período de estacionalidad especificado. Dado que la descomposición estacional del apartado 3.2.2 mostró una fuerte estacionalidad semanal en los datos, se encontró que valores altos de γ producen mejores predicciones, como hemos visto en los ejemplos anteriores.

Simulated Annealing

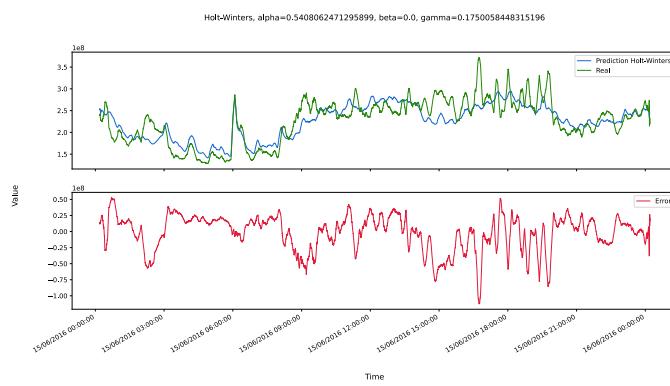


Figura 3.9: Predicción de los datos de tráfico normal con Holt-Winters, con MSE de $1,85 \cdot 10^{15}$

Se aplicó *Simulated Annealing* para minimizar el MSE en la predicción. Este enfoque logró reducir ligeramente el MSE para una ventana específica de predicción. Sin embargo, los mismos parámetros no se ajustaron bien para otras ventanas de predicción, mostrando cierta limitación en la generalización de los resultados. En la figura 3.9, se muestra un ejemplo de predicción con un MSE de $1,85 \cdot 10^{15}$. Sin embargo, este enfoque no ha sido exitoso en general, ya que no ha logrado reducir el MSE de manera significativa en todas las ventanas de predicción.

3.3. Simulación de ataque

Para simular un ataque de denegación de servicio, hemos generado un tráfico anómalo que consiste en un aumento del *bitrate* durante un periodo de 2 minutos. Este ataque se puede simular en cualquier ventana que seleccionemos para analizar. Además, se plantean dos escenarios de ataque:

- **Escenario 1:** El ataque produce un pico de tráfico en una ventana de tráfico normal.
- **Escenario 2:** El ataque produce un incremento gradual del tráfico en una ventana de tráfico normal.

En la figura 3.10, se muestra en color rojo la simulación de un ataque de denegación de servicio en la ventana de las 10:00 a las 10:15 del día 6 de junio de 2016. En este caso, se observa un pico de

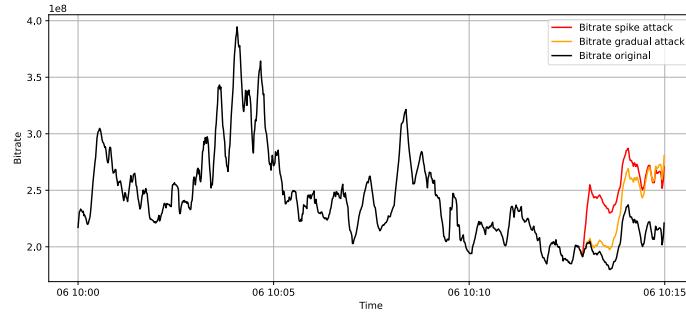


Figura 3.10: Simulación de un ataque de denegación de servicio en el *bitrate* en una ventana de 15 minutos

tráfico en el *bitrate* aumenta el tráfico en 50 Mbps durante 2 minutos. En cambio, en color naranja, se muestra la simulación de un ataque correspondiente al escenario 2, aplicando un incremento gradual al *bitrate*. En este caso se alcanza el máximo al final de la ventana, llegando a 60 Mbps en 2 minutos.

El caso del escenario 2 es de especial interés, ya que es más difícil de detectar por su naturaleza, ya que el incremento es gradual y no produce un pico de tráfico tan evidente como en el escenario 1. Por tanto, las pruebas de detección de anomalías se centrarán en este escenario.

3.4. Conclusión

En el análisis de datos hemos realizado la descomposición estacional de los datos y hemos observado su comportamiento general. Hemos identificado patrones significativos en los datos y destacado el uso de Holt-Winters para el análisis de series temporales con estacionalidad. También hemos sugerido dos simulaciones de ataques de denegación de servicio.

El escenario de ataque más sutil y, por tanto, de mayor interés se pondrá a prueba para detectar las anomalías que genera. Diseñaremos un sistema con el que se podrán detectar con alta precisión los ataques.

DISEÑO DEL SISTEMA

En este capítulo se presenta el diseño del sistema de detección de anomalías en el tráfico de red. En primer lugar, se describe el proceso de estacionarización de los datos, que es necesario para poder aplicar los métodos de detección de anomalías. A continuación, se presenta el esquema de funcionamiento del sistema, que se basa en la detección de anomalías mediante la estandarización de los datos. Por último, se presentan las conclusiones de este capítulo.

4.1. Estacionarización de los datos

Como vimos en la sección 3.2.1, los datos de entrada no pueden ser considerados estacionarios, por lo que es necesario realizar un ajuste a los mismos para poder trabajar con ellos y aplicar los métodos de predicción. Mostramos en la figura 4.1 el trabajo que realizaremos para obtener una serie temporal estacionaria. En esta figura también señalamos el análisis previo realizado que ya ha culminado, como la descomposición estacional o la predicción del tráfico mediante el suavizado exponencial Holt-Winters.

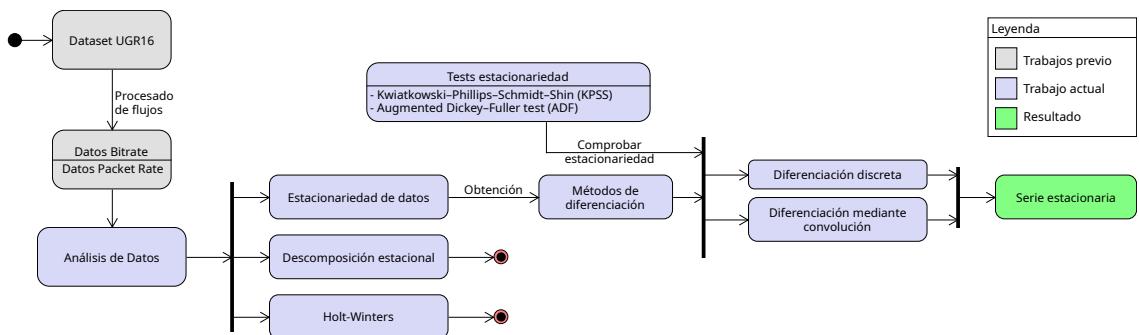


Figura 4.1: Proceso de la estacionarización de los datos

4.1.1. Diferenciación de los datos

La diferenciación es una técnica común utilizada en el análisis de series temporales para hacer que los datos sean estacionarios [19]. Consiste en calcular diferencias entre observaciones sucesivas en la serie temporal. En este contexto, la diferenciación discreta es particularmente útil para eliminar tendencias y hacer que la serie sea estacionaria.

La diferenciación discreta de orden 1 se define como:

$$\Delta y_t = y_t - y_{t-1} \quad (4.1)$$

donde y_t es el valor en el tiempo t de la serie temporal. Esta técnica ayuda a eliminar las tendencias lineales y puede ser efectiva para hacer que la serie sea estacionaria.

Test	P-valor medio	Porcentaje de ventanas no estacionarias
ADF	$2,42 \cdot 10^{-12}$	0,00 %
KPSS	0,099	0,80 %

Tabla 4.1: Estacionariedad de la serie con diferenciación discreta

En nuestro estudio, hemos encontrado que una diferenciación de primer orden es suficiente para lograr la estacionarización de los datos. No es necesario aplicar diferenciaciones de orden superior dado que los datos se vuelven estacionarios después de una única diferenciación. En la tabla 4.1 se muestran los resultados de las pruebas de estacionariedad aplicadas a todas las ventanas de los datos, aplicando el método de diferenciación discreta. Se observa que el p-valor medio de la prueba KPSS es cercano a 0,1, y no superior a eso. Esto se debe a una limitación de la implementación del test KPSS en Python, que calcula el p-valor de manera aproximada mediante una tabla de consulta.

Además de la diferenciación discreta, también hemos explorado una técnica alternativa utilizando una convolución con el siguiente *kernel*:

$$\left[-\frac{1}{2}, 0, \frac{1}{2} \right] \quad (4.2)$$

Esta convolución también es efectiva para eliminar tendencias lineales y puede ser útil para suavizar la serie mientras se eliminan las tendencias.

Test	P-valor medio	Porcentaje de ventanas no estacionarias
ADF	$2,45 \cdot 10^{-10}$	0,00 %
KPSS	0,099	0,4 %

Tabla 4.2: Estacionariedad de la serie con diferenciación mediante convolución

En la tabla 4.2 se muestran los resultados de las pruebas de estacionariedad aplicadas a todas las ventanas de los datos, aplicando el método de diferenciación mediante convolución. En este caso

también podemos afirmar que prácticamente todas las ventanas son estacionarias después de aplicar la convolución.

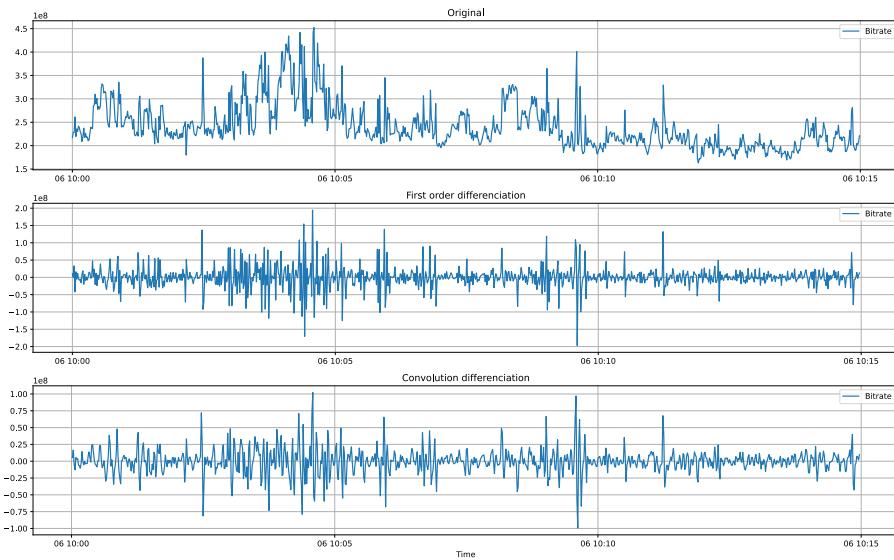


Figura 4.2: Diferenciación de los datos mediante diferenciación discreta y convolución

En la figura 4.2 se muestra el resultado de la diferenciación de los datos mediante la técnica de diferenciación discreta y convolución. Se puede observar que ambas técnicas son efectivas para hacer que los datos sean estacionarios.

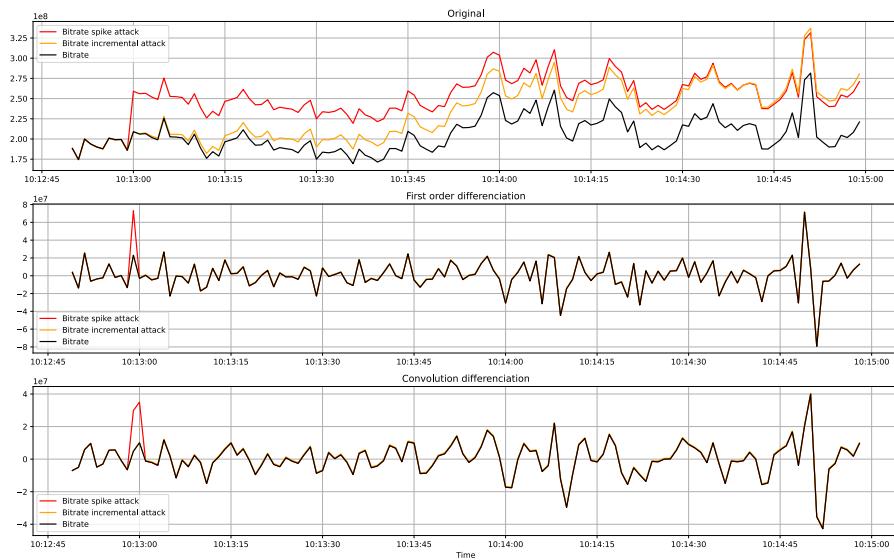


Figura 4.3: Diferenciación de los datos sujetos a un ataque

Si aplicamos un ataque a esta ventana, como se muestra en la figura 4.3, podemos ver que la diferenciación de los datos sigue siendo efectiva para hacer que la serie sea estacionaria. En particular, observamos que un ataque con pico produce otro pico en la diferenciación. Sin embargo, un ataque gradual parece pasar desapercibido en ambas técnicas de diferenciación. Además, se observa cómo

el ataque con pico se extiende durante más tiempo en la serie en la cual se aplica la convolución. Esta propiedad puede ser de utilidad para detectar ataques en la serie temporal, pues los efectos de los ataques se extienden en el tiempo.

4.2. Esquema de funcionamiento

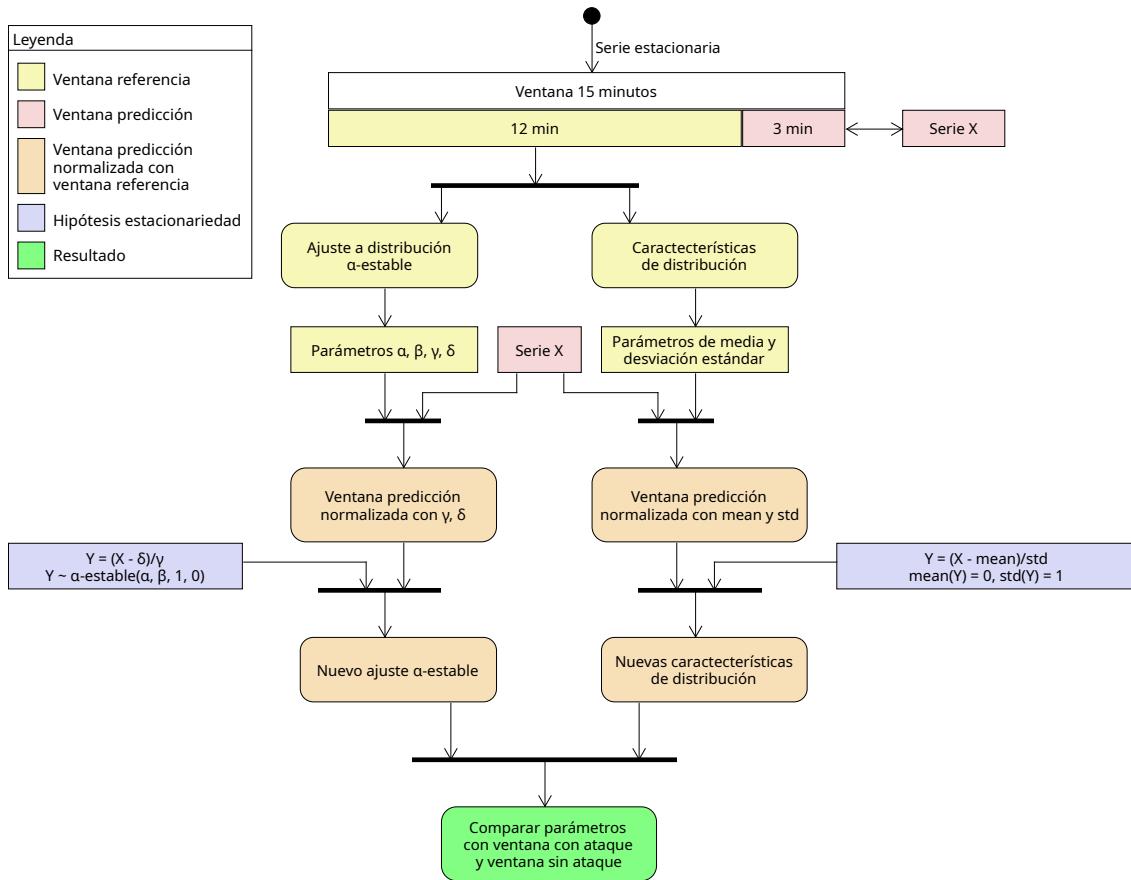


Figura 4.4: Proceso completo de detección de anomalías en el tráfico de red

Hemos obtenido la estacionariedad de los datos mediante dos métodos de diferenciación en la sección 4.1. Ahora vamos a introducir el tratamiento de datos propuesto por B. Martín en su estudio [4], y otro método novedoso de normalización de los datos, para poder detectar una posible anomalía en el tráfico de red. En la figura 4.4 se muestra el proceso de detección de anomalías en el tráfico de red, que parte de una ventana de 15 minutos de tráfico de red, y se divide en dos ramas: una rama que ajusta los datos a una distribución α -estable, y otra rama que normaliza los datos mediante la tipificación. Este proceso se detalla en los siguientes apartados.

Cabe destacar que la elección del tamaño de ventana de 15 minutos se debe a que es un tiempo suficiente para detectar anomalías en el tráfico comparando con el comportamiento previo. Estos 15 minutos de tamaño de ventana equivalen a 900 puntos de datos (1 por segundo), lo que nos permite

realizar los ajustes de los apartados siguientes de manera eficiente.

4.2.1. Ajuste α -estable

El ajuste de los datos a una distribución α -estable es una parte esencial del análisis propuesto por B. Martín [4]. Su hipótesis es que una serie estacionaria sigue un proceso de una distribución α -estable, lo que permite modelar los datos y explorar su comportamiento bajo esta distribución. De esta forma, podemos ajustar los datos estacionarizados a una distribución α -estable y podemos tomarlo como distribución base, con sus respectivos parámetros α (estabilidad), β (asimetría), γ (escala) y δ (localización). Luego se puede comparar el ajuste de otros datos con la distribución base, y detectar si hay alguna anomalía en el tráfico de red, suponiendo que la distribución base no está sujeta a un ataque.

Para realizar este ajuste, utilizamos la función `fitdist` en MATLAB con el parámetro '`Stable`'. Este método es computacionalmente eficiente, y nos permite modelar la distribución de los datos y explorar su comportamiento bajo esta distribución para una gran cantidad de datos.

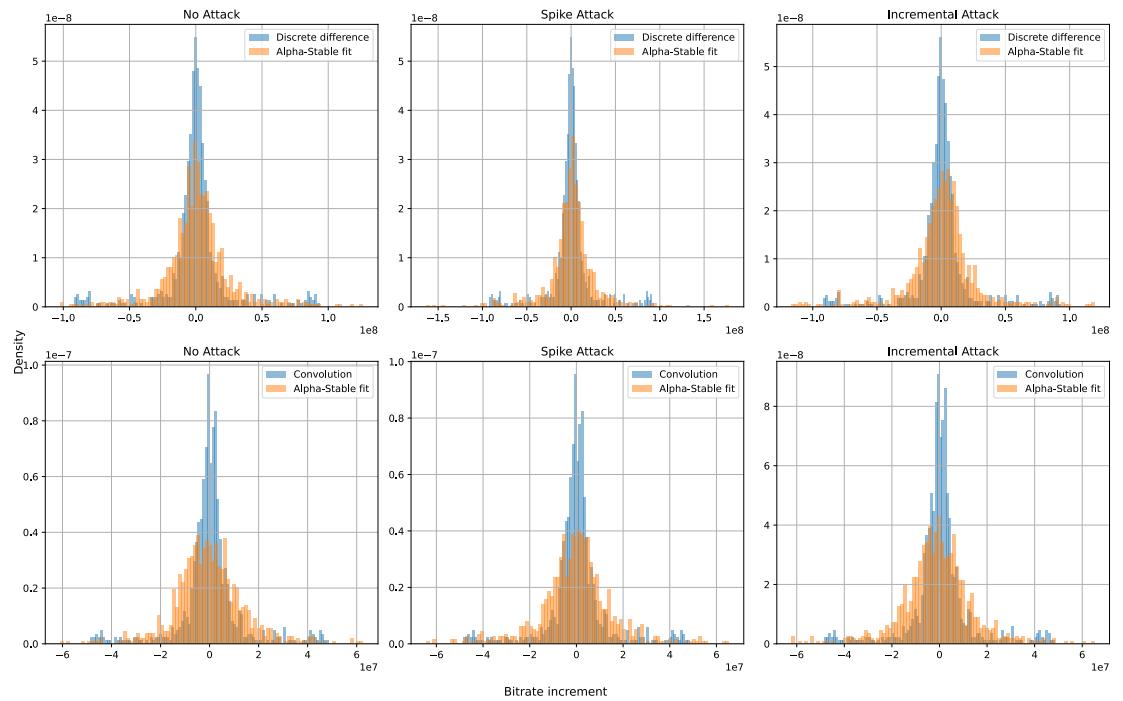


Figura 4.5: Ajuste de los datos a una distribución α -estable

En la figura 4.5 se muestra el ajuste de los datos a una distribución α -estable. Se realiza el ajuste de los datos diferenciados con ambos métodos (diferencias discretas de primer orden y convolución), bajo los escenarios sin ataque, con un ataque de pico y un ataque incremental. En azul se muestran los datos reales, y en naranja un muestreo de la distribución α -estable ajustada. Este muestreo tiene el mismo número de observaciones que la serie diferenciada. Se observa que la escala de los datos es

muy amplia, por el orden de 10^7 , por lo que el ajuste no es óptimo. Por ello propondremos un método de normalización de los datos en la siguiente sección.

4.2.2. Normalización de los datos

Como hemos indicado previamente, proponemos otro método para comparar los datos del tráfico de red en ventanas fijas, mediante la normalización de los datos. La normalización es un proceso que permite escalar los datos para que se encuentren dentro de un rango específico, facilitando la comparación entre diferentes conjuntos de datos.

En este apartado vamos a explorar dos métodos de detección de anomalías. Uno de ellos consiste en el ajuste de los datos a una distribución α -estable, propuesto por Benjamín Martín [4]. El otro método consiste en la comparación de la tipificación de diferentes ventanas de tráfico.

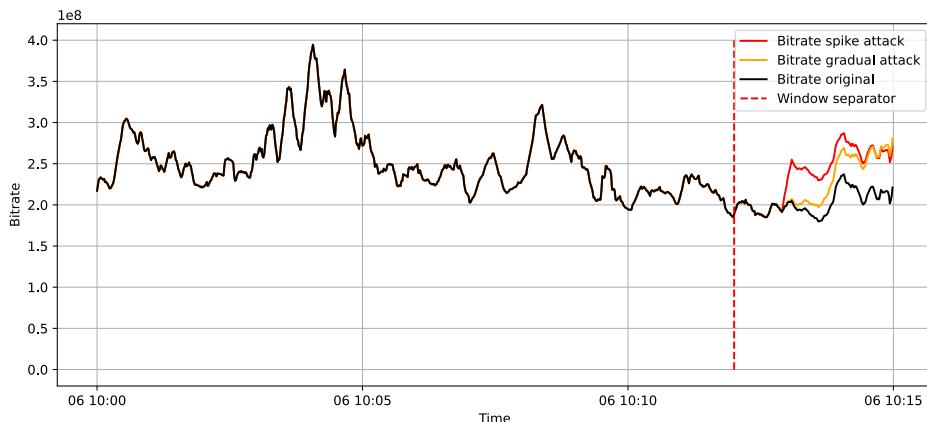


Figura 4.6: Ventana ejemplo de 15 minutos sobre la que se realiza el análisis

En la figura 4.6 se muestra una ventana de 15 minutos sobre la que se realiza el análisis. Este análisis tendrá en consideración los primeros 12 minutos del tráfico como ventana de referencia, para luego predecir sobre los 3 minutos siguientes si se produce alguna anomalía. En la figura se separan las ventanas de referencia y de predicción con una línea vertical punteada. El objetivo es detectar cualquier discrepancia significativa en el comportamiento del tráfico de red en los 3 minutos siguientes, en comparación con los 12 minutos anteriores.

Normalizando los datos con la distribución α -estable

Como vimos en la figura 4.5, los datos se ajustan bien a una distribución α -estable. Sin embargo, los datos no están normalizados, lo que dificulta la comparación de los datos. Por ello, normalizamos los datos con la distribución α -estable ajustada, como explica J.P. Nolan en el libro *Univariate Stable Distributions* [9]. Sea X una variable aleatoria con distribución α -estable $S(\alpha, \beta, \gamma, \delta)$, entonces la

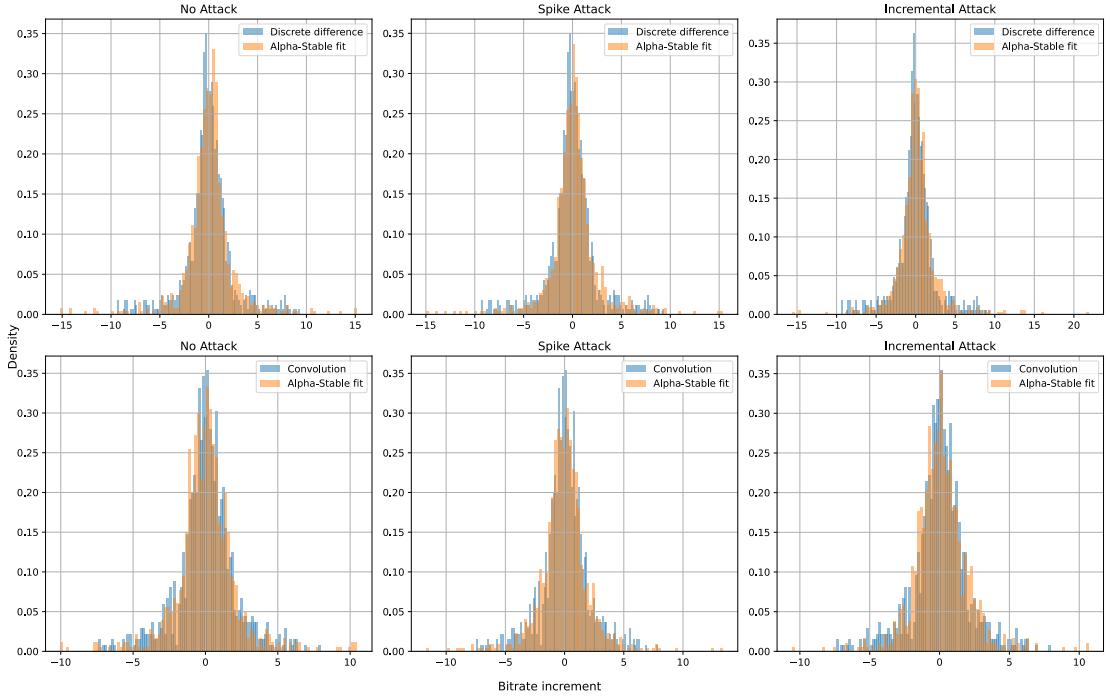


Figura 4.7: Ajuste de los datos a una distribución α -estable normalizando los datos

variable aleatoria $Y = (X - \delta)/\gamma$ tiene distribución α -estable $S(\alpha, \beta, 1, 0)$.

Aplicando este método de normalización, repetimos el ajuste de distribución α -estable a los datos normalizados. En la figura 4.7 se muestra el ajuste de los datos normalizados a una distribución α -estable. Al igual que en la figura 4.5, se realiza el ajuste de los datos diferenciados con los métodos de diferenciación de primer orden y mediante convolución, bajo los escenarios sin ataque, con un ataque de pico y un ataque incremental. En azul se muestran los datos reales normalizados, y en naranja un muestreo de la distribución α -estable ajustada normalizada.

Método	Sin ataque	Con ataque de pico	Con ataque incremental
Diferenciación de primer orden	$\alpha = 1,2267$ $\beta = -0,0214$	$\alpha = 1,2218$ $\beta = -0,0305$	$\alpha = 1,2269$ $\beta = -0,0200$
Convolución	$\alpha = 1,3796$ $\beta = 0,0306$	$\alpha = 1,3732$ $\beta = 0,0287$	$\alpha = 1,3796$ $\beta = 0,0330$

Tabla 4.3: Ajuste de los datos a una distribución α -estable normalizando los datos, $\gamma = 1$ y $\delta = 0$

En la tabla 4.3 se muestran los parámetros α y β de la distribución α -estable ajustada a los datos normalizados. Los parámetros γ y δ son fijos en 1 y 0 respectivamente, gracias a la normalización.

Ahora que los datos están normalizados, podemos comparar los ajustes α -estables de diferentes ventanas de tráfico para detectar posibles anomalías en el tráfico de red. El análisis propuesto consiste en comparar el ajuste α -estable de una ventana de 12 minutos con el ajuste α -estable de los 3 minutos

siguientes, con el objetivo de identificar cualquier discrepancia significativa en el comportamiento del tráfico.

La idea detrás de esta comparación es verificar si existe una diferencia sustancial en la distribución α -estable entre las dos ventanas de tiempo consecutivas. Si observamos cambios abruptos o anómalos en los parámetros α , β , γ , o δ entre estas dos ventanas, podría indicar la presencia de una anomalía en el tráfico de red durante ese período.

Normalizando los datos mediante la tipificación

Otro método de normalización de los datos es la tipificación. La tipificación es un método comúnmente utilizado en el análisis de series temporales para hacer que los datos sean comparables. Consiste en restar la media y dividir por la desviación estándar de los datos. En este contexto, la tipificación es particularmente útil para hacer que los datos sean comparables y explorar su comportamiento.

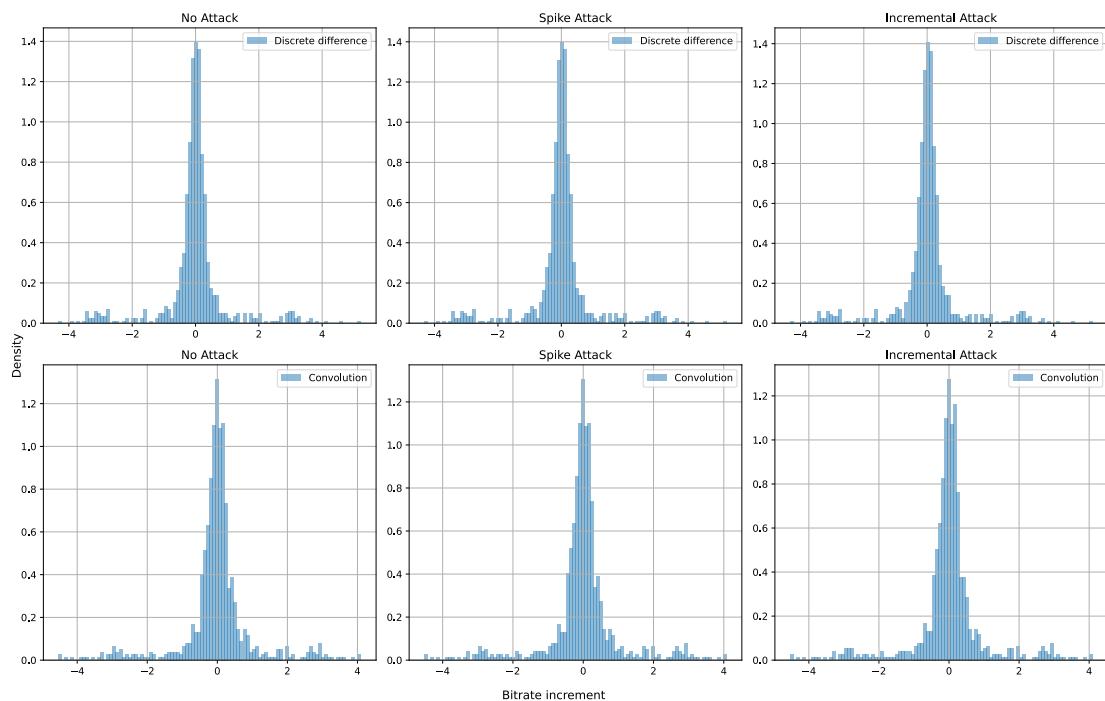


Figura 4.8: Normalización de los datos mediante la tipificación

En la figura 4.8 se muestra la normalización de los datos mediante la tipificación. Se observa que los datos estandarizados tienen una media de 0 y una desviación estándar de 1, lo que facilita la comparación de los datos. Al igual que en el apartado 4.2.2, podemos comparar la tipificación de diferentes ventanas de tráfico para detectar posibles anomalías en el tráfico de red.

Como se ha comprobado previamente la estacionariedad de los datos de la ventana de 15 minutos, podemos afirmar que sigue un proceso estacionario, es decir, que la media y la varianza de los datos no cambian con el tiempo. Por tanto, podemos asumir que, si no hay ningún ataque en el tráfico de red, la distribución de los datos de la ventana de 12 minutos debería ser la misma que la de los 3 minutos

siguientes. Bajo estas condiciones, proponemos el siguiente proceso de detección de anomalías:

- 1.– Obtener la media μ y desviación estándar σ de la ventana de 12 minutos.
- 2.– Tipificar la ventana de 3 minutos con la media y desviación estándar obtenidas en el paso anterior.
- 3.– Comparar la tipificación de los datos de la ventana de 3 minutos con una distribución normal estándar.
 - 3.1.– Bajo la hipótesis de que no hay ataque en el tráfico de red, la tipificación de los datos de la ventana de 3 minutos debería seguir una distribución normal estándar, es decir, una distribución con media 0 y desviación estándar 1.
 - 3.2.– Si la media y/o la desviación estándar de la tipificación de los datos de la ventana de 3 minutos difieren significativamente de 0 y 1 respectivamente, podríamos concluir que hay una anomalía en el tráfico de red durante ese período. En este caso, hay que definir un umbral para determinar cuándo una diferencia es significativa.

4.3. Conclusión

En este capítulo hemos presentado una metodología para la detección de anomalías en el tráfico de red que se basa en la estandarización de los datos. Esta metodología necesita que los datos de entrada sean estacionarios, por lo que hemos aplicado técnicas de diferenciación y convolución para garantizar la estacionariedad de los datos. Hemos encontrado que una diferenciación de primer orden es suficiente para lograr la estacionarización de los datos. Además, hemos explorado una técnica alternativa utilizando una convolución con un *kernel* específico. Ambas técnicas son efectivas para hacer que los datos sean estacionarios, pero la convolución tiene la propiedad de extender los efectos de los ataques en la serie temporal.

Con estos métodos en mente, podemos aplicar la metodología de detección de anomalías sobre los datos estacionarizados. En el siguiente capítulo, analizaremos los resultados obtenidos y evaluaremos la efectividad de nuestra estrategia para detectar y caracterizar anomalías en el tráfico de red.

RESULTADOS OBTENIDOS

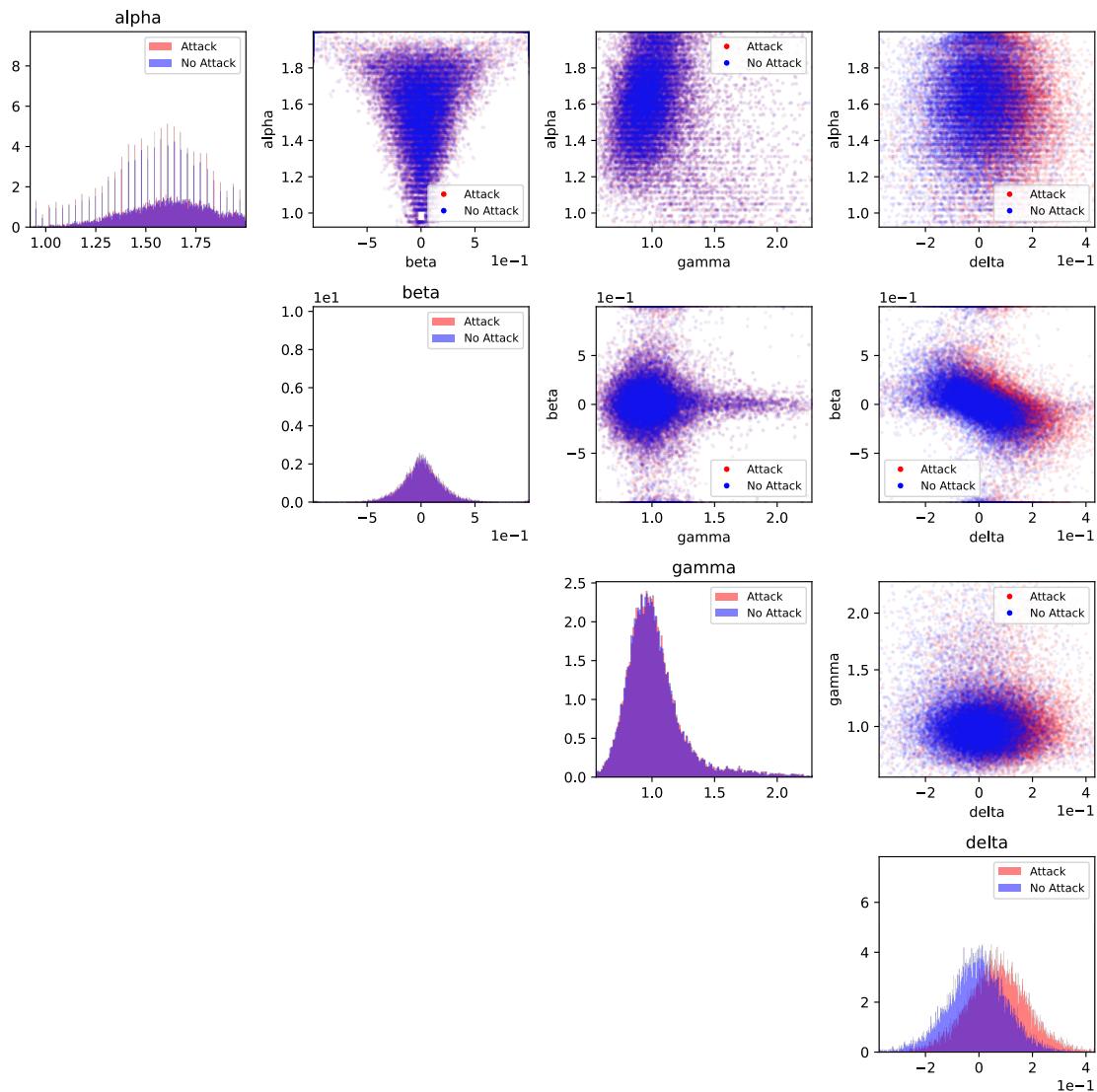
En esta sección se describen las pruebas realizadas para evaluar el rendimiento del sistema de detección de anomalías. Se aplicarán las pruebas descritas en el capítulo 4 para todas las ventanas posibles. Las ventanas que se consideran tienen un tamaño de 15 minutos y se desplazan cada 1 minuto a lo largo de la serie temporal, lo que permite una evaluación detallada del sistema de detección. Sin embargo, el tipo de ataque se ha limitado a un ataque de tráfico incremental, ya que es el más difícil de percibir y, por tanto, el más interesante de detectar. Un ataque con un aumento de tráfico en pico es más fácil de detectar, ya que se puede observar un aumento brusco en el tráfico, como vimos en la sección 4.1, en la figura 4.3. Además, el tipo de diferenciación que se aplica es mediante convoluciones, ya que extienden los efectos de los ataques en la serie temporal.

En las pruebas se analizarán los falsos positivos y falsos negativos obtenidos, para determinar la eficacia del sistema. También se aplicarán métodos para mejorar las predicciones, como regresión logística y SVM.

5.1. Detección mediante α -estables

Para la detección de anomalías mediante la comparación de distribuciones α -estables, se han computado los ajustes de las distribuciones de las subventanas de 12 minutos tomadas como referencia. Como se describía en el apartado 4.2.2, se ha normalizado los datos de la subventana de 3 minutos sujeta a un posible ataque. Posteriormente, se ha ajustado la distribución de los datos normalizados de la subventana de 3 minutos.

En la figura 5.1 se muestra la distribución de los valores de α , β , γ y δ obtenidos en el ajuste de las subventanas de 3 minutos como se ha explicado. Los puntos o las barras de color rojo indican que la ventana estaba sujeta a un ataque, mientras que en color azul se indica que no lo estaba. En diagonal observamos los histogramas de los parámetros, donde se muestra la distribución que siguen los valores obtenidos según estuvieran bajo ataque. En el resto de posiciones se muestran diagramas de dispersión entre dos parámetros. En los valores de α se observan múltiples valores que se repiten a menudo. Esto se debe a un artefacto en el cómputo del ajuste α -estable.

**Figura 5.1:** Distribución de los valores del ajuste α -estable

	Media	Desviación estándar
Ataque	-0,0740	15,3869
Sin ataque	-0,0043	3,4001

Tabla 5.1: Características de la distribución del parámetro δ

Destaca en esta figura cómo en la diagonal los valores de α , β y γ siguen la misma distribución cuando hay un ataque que cuando no lo hay, mientras que la distribución del parámetro δ sí que varía. Esto es de esperar, ya que δ es el parámetro de localización y describe en qué valor está centrada la función densidad de probabilidad. Por tanto, si hay un ataque que supone un aumento en el tráfico, el centro de la distribución de los datos se desplazará a la derecha, lo que se reflejará en el parámetro δ . En la tabla 5.1 vemos que la media de los valores de δ está alrededor del 0 independientemente de si hay un ataque o no, pero la desviación estándar de los valores de δ es mucho mayor cuando hay un ataque que cuando no lo hay.

Los diagramas de dispersión muestran solapamiento entre los valores de los parámetros α , β y γ cuando hay un ataque y cuando no lo hay, por lo que estos parámetros no son útiles para la detección de anomalías. Por el contrario, el parámetro δ muestra una clara separación con los demás valores. Sin embargo, esta separación parece ser un desplazamiento de la distribución de los valores, y no una diferencia en la forma de la distribución. Por tanto, parece razonable utilizar únicamente el parámetro δ para la detección de anomalías.

5.2. Detección mediante la media

De forma similar a la detección de anomalías mediante la comparación de distribuciones α -estables, dada una ventana de 15 minutos, se han tipificado los datos de la subventana de 3 minutos en la que se quiere detectar un posible ataque. Dicho proceso de tipificación se ha realizado con la media y desviación estándar de la subventana de 12 minutos tomada como referencia, como se indica en la sección 4.2.2. Posteriormente, se ha calculado la media y desviación estándar de los datos tipificados de la subventana de 3 minutos. Para evitar confusión en la lectura, estos valores de media y desviación estándar se referirán como parámetros *mean* y *std* respectivamente.

En la figura 5.2 se muestra la distribución de los parámetros *mean* y *std* obtenidos en la tipificación de las subventanas de 3 minutos. En color rojo indican que la ventana estaba sujeta a un ataque, mientras que en color azul se indica que no lo estaba. Al igual que en la figura 5.1, en diagonal observamos los histogramas de los parámetros, donde se muestra la distribución que siguen los valores obtenidos según estuvieran bajo ataque.

	Media	Desviación estándar
Ataque	0,0339	0,0270
Sin ataque	0,0005	0,0235

Tabla 5.2: Características de la distribución del parámetro *mean*

En el histograma de *mean* se observa una clara diferenciación entre las distribuciones de los valores obtenidos cuando hay un ataque y cuando no lo hay; en la localización de la distribución. Como

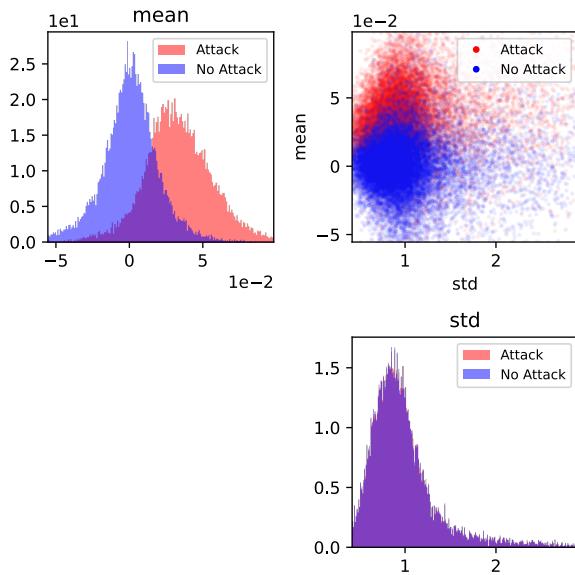


Figura 5.2: Distribución de los parámetros *mean* y *std* tras la tipificación

vemos en la tabla 5.2, la distribución de los valores de *mean* cuando no hay ataque se encuentra en torno a 0, mientras que cuando hay un ataque se encuentra en torno a 0,34. Además, la desviación estándar de los valores de *mean* es relativamente pequeña, lo que nos permite diferenciar entre ambas distribuciones.

Por el contrario, en el histograma de *std* no se observa ninguna diferencia entre las distribuciones que siguen estos valores. En el diagrama de dispersión se observa únicamente un desplazamiento de los puntos en el eje correspondiente a *mean*, lo que indica que *std* no es útil para la detección de anomalías. Por tanto, parece razonable considerar únicamente el parámetro *mean* para la detección de anomalías.

5.3. Detección combinada

Como se ha encontrado que tanto el parámetro *mean* como el parámetro δ son relevantes en cuanto a la detección de anomalías, se plantea combinar ambos métodos para mejorar la detección de anomalías.

	δ	Media
δ	1,000000	0,003454
Media	0,003454	1,000000

Tabla 5.3: Correlación entre los valores de δ y *mean*

En la figura 5.3 se muestra la distribución de los valores de δ y *mean* obtenidos en los apartados

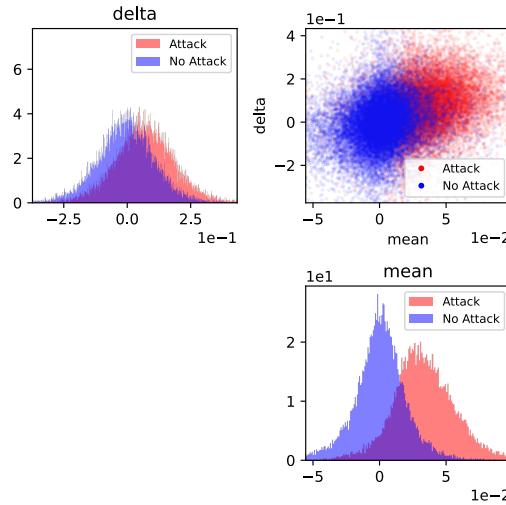


Figura 5.3: Distribución de los parámetros δ del ajuste α -estable y $mean$

anteriores. Se ha calculado la matriz de correlación entre ambos parámetros, que se muestra en la tabla 5.3. Se observa que la correlación entre ambos parámetros es muy baja, por lo que se plantea que la combinación de ambos métodos puede ser útil para la detección de anomalías.

5.4. Métricas de evaluación

Por las características de los datos que hemos observado, se plantea que la combinación de ambos métodos puede ser útil para la detección de anomalías. Por ello, se propone un análisis de regresión logística y SVM, utilizando como variables independientes el parámetro $mean$ y δ obtenidos en el ajuste de la distribución α -estable. SVM es útil para separar eficazmente datos en diferentes clases mediante la identificación de un hiperplano óptimo en un espacio dimensional superior.

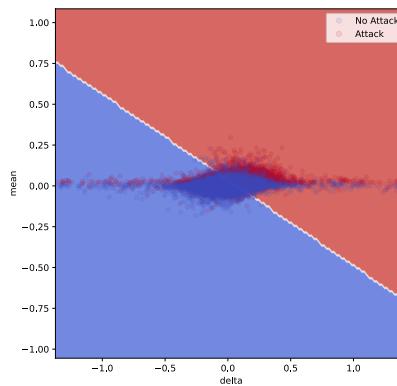
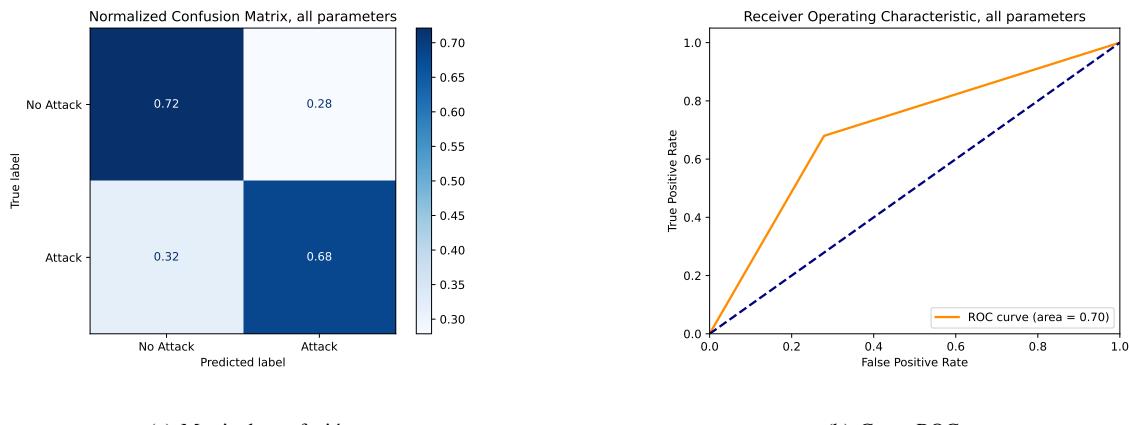


Figura 5.4: SVM de los parámetros δ y $mean$, utilizando kernel RBF



(a) Matriz de confusión

(b) Curva ROC

Figura 5.5: Matriz de confusión y curva ROC del SVM de los parámetros δ y $mean$

En SVM, el *kernel* Función de Base Radial, *Radial Basis Function* (RBF) se utiliza comúnmente para abordar problemas de clasificación no lineales, ya que permite capturar relaciones más complejas entre las características de entrada. En la figura 5.4 se muestra el resultado de la clasificación de los parámetros δ y $mean$ utilizando un *kernel* RBF. Se observa que la clasificación obtenida sigue un patrón lineal, por lo que se plantea que un *kernel* lineal puede ser suficiente para clasificar los datos. En este caso, la regresión logística puede ser una alternativa más sencilla y eficiente (y en algunos casos mejor [20]) para clasificar los datos.

Aplicando SVM, se ha obtenido una exactitud o *accuracy* del 70,05 %, una precisión del 70,09 %, una sensibilidad o *recall* del 67,98 % y un *F1 score* del 69,41 %. En la figura 5.5 observamos la matriz de confusión del SVM con los parámetros δ y $mean$ y su curva Característica Operativa del Receptor, *Receiving Operating Characteristic* (ROC).

	coef	std err	z	P> z
const	-1,0319	0,018	-57,341	0,000
δ	-0,0010	0,002	-0,644	0,519
mean	62,9153	0,744	84,588	0,000

Tabla 5.4: Resultados de la regresión logística con parámetros δ y $mean$

En la tabla 5.4 se muestran los resultados de la regresión logística. Para poner a prueba la eficacia de la regresión logística, se ha realizado una validación cruzada con un 80 % de los datos para entrenamiento y un 20 % para validación. Se ha procurado que las ventanas que se usan para entrenamiento y *test* no se solapen en ningún punto, ya que esto podría introducir sesgos en el modelo. Esto se ha conseguido eliminando los datos de validación que comparten algún punto con los datos de entrenamiento.

En la figura 5.6 se muestra la matriz de confusión de la regresión logística con los parámetros δ

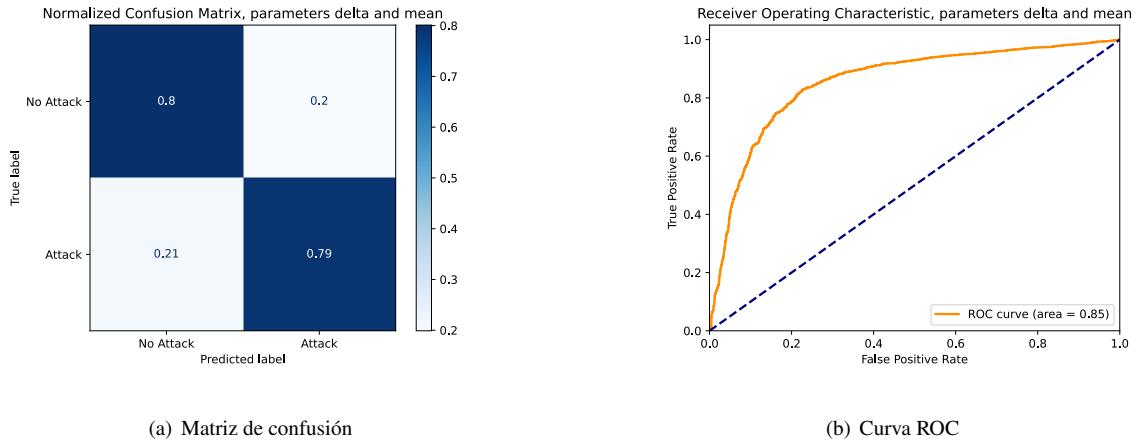


Figura 5.6: Matriz de confusión y curva ROC de la regresión logística con parámetros δ y $mean$

y $mean$ y su curva ROC. Se ha obtenido un *accuracy* del 79,4 %, una precisión del 79,8 %, un *recall* del 78,7 % y un *F1 score* del 79,3 %. Estos resultados indican que la regresión logística es capaz de clasificar correctamente la mayoría de los datos, aunque se observa que hay un número significativo de falsos positivos y falsos negativos. Por ello, se plantea que la regresión logística puede ser útil para clasificar los datos, pero no es suficiente para obtener una clasificación perfecta.

	coef	std err	z	P> z
const	-0,1488	0,122	-1,218	0,223
α	-0,4842	0,069	-7,064	0,000
β	-0,3669	0,039	-9,426	0,000
γ	0,4641	0,065	7,130	0,000
δ	-0,0020	0,002	-1,003	0,316
mean	64,9292	0,769	84,453	0,000
std	-0,6110	0,051	-11,943	0,000

Tabla 5.5: Resultados de la regresión logística con todos los parámetros

También se ha realizado un análisis de regresión logística con todos los parámetros considerados. Los resultados de la regresión logística se muestran en la tabla 5.5. En la figura 5.7 se muestra la matriz de confusión de la regresión logística con todos los parámetros y su curva ROC. Se ha obtenido un *accuracy* del 83,1 %, una precisión del 82,2 %, una sensibilidad del 84,6 % y un *F1 score* del 83,4 %. Estos resultados indican que la regresión logística con todos los parámetros es más eficaz que la regresión logística que solo usa los parámetros δ y $mean$, ya que se obtiene una mejora en todas las métricas de evaluación.

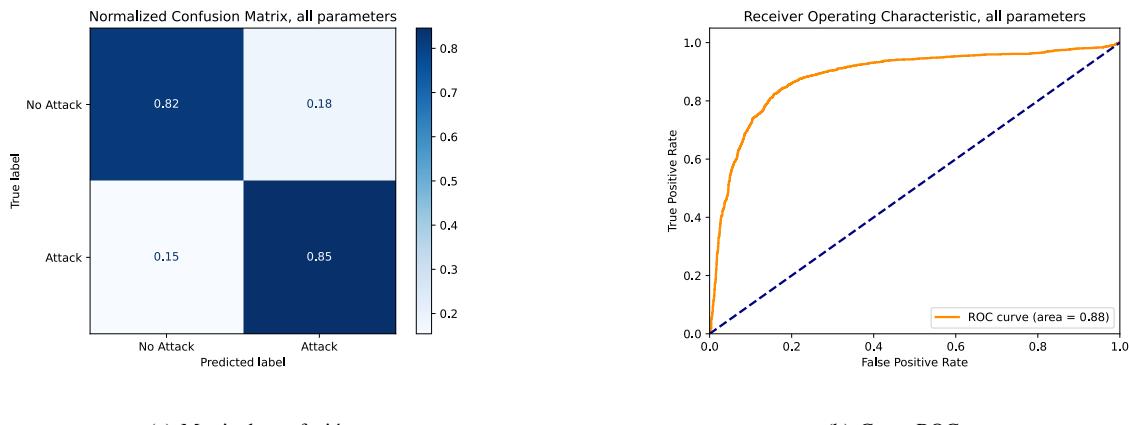


Figura 5.7: Matriz de confusión y curva ROC de la regresión logística con todos los parámetros

5.4.1. Detección sobre el conjunto de datos de prueba

Un resultado de interés es comprobar en qué ventanas del dataset se pueden detectar anomalías y en qué ventanas se puede confirmar que no hay anomalías correctamente. Para ello, se ha realizado una predicción utilizando el método de regresión logística con todos los parámetros, ya que obtenemos los mejores resultados con este método. Se ha utilizado el 80 % de los datos para el entrenamiento de la regresión, y se ha realizado la prueba sobre los datos restantes, tras eliminar los datos que puedan solaparse con los datos de entrenamiento.

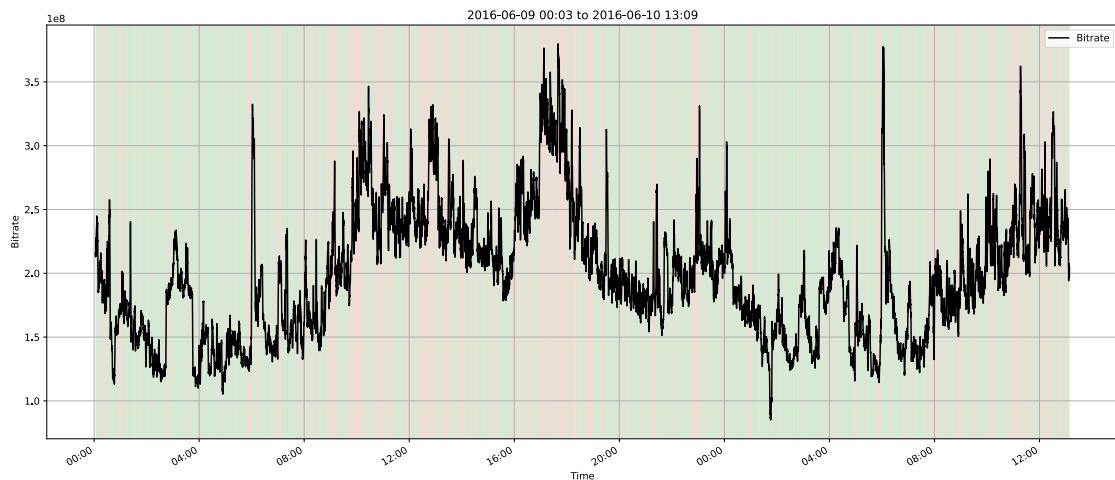


Figura 5.8: Clasificación de las ventanas del dataset sujeto a ataques, entre los días 09/06/2016 (jueves) y 10/06/2016 (viernes), utilizando regresión logística con todos los parámetros

En la figura 5.8 se muestra la clasificación de las ventanas del dataset utilizando regresión logística con todos los parámetros. Para una representación más clara, se ha utilizado una ventana móvil de 60 segundos. En rojo se señalan las ventanas incorrectamente clasificadas como no ataques, mientras

que en verde se destacan las ventanas correctamente clasificadas como ataques. La mayoría de las ventanas están clasificadas correctamente, pero hay un número considerable de falsos negativos que tienden a concentrarse alrededor de ciertas áreas. El porcentaje de falsos negativos es del 21,87 %. Se aprecia que los falsos negativos suceden en ventanas donde hay mucha variabilidad del tráfico. En este caso sucede en los períodos entre las 09:00 y 11:00 y también entre las 16:00 y 19:00 del jueves, 09, de junio de 2016. Estas clasificaciones incorrectas se deben a que el tráfico en estos períodos se asemeja al tráfico de ataque, mostrando una carga incremental, por lo que el modelo presenta más dificultades para clasificar correctamente estas ventanas.

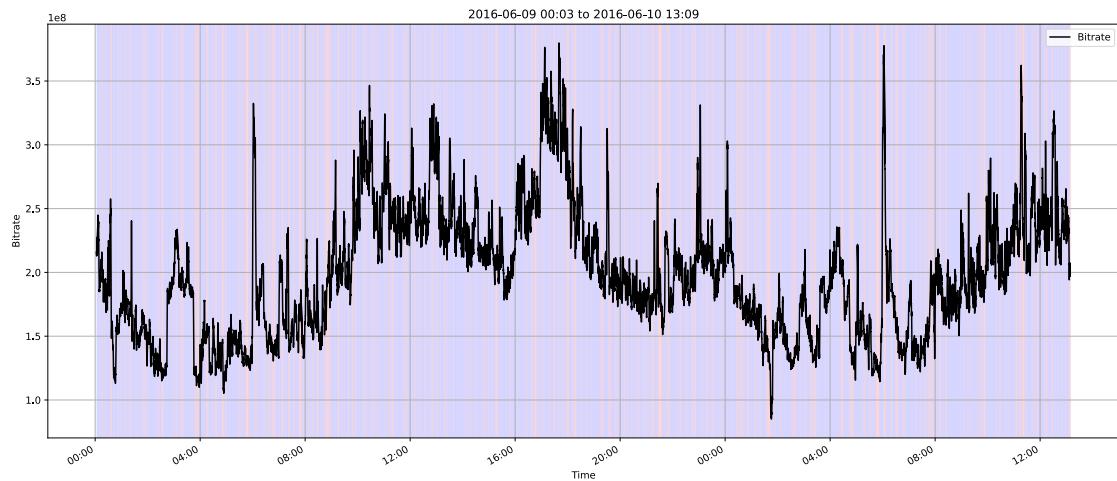


Figura 5.9: Clasificación de las ventanas del dataset sin aplicar ataque, entre los días 09/06/2016 (jueves) y 10/06/2016 (viernes), utilizando regresión logística con todos los parámetros

En la figura 5.9 se muestra la clasificación de las ventanas del dataset utilizando regresión logística con todos los parámetros. Se ha utilizado una ventana móvil de 60 segundos para una representación más clara. En color rojo se indican las ventanas que se han incorrectamente clasificadas como ataque, mientras que en color azul se indican las ventanas que han sido correctamente clasificadas como no ataque. Al igual que en el caso anterior, la mayoría de las ventanas han sido clasificadas correctamente, y el porcentaje de falsos positivos es del 17,24 %. En este caso, no se observan concentraciones de falsos positivos, sino que se distribuyen de forma más uniforme a lo largo del día, por lo que podemos concluir que el método no está influenciado por ventanas con mucho o poco tráfico.

5.5. Conclusión

En esta sección hemos explorado la utilidad de los parámetros derivados de ajustes de distribuciones α -estables y la tipificación de los datos introducidos en el capítulo 4. Descubrimos que el parámetro *mean* es efectivo para la detección de anomalías por sí mismo. También observamos que la combinación de ambos parámetros puede mejorar aún más la detección de anomalías.

CONCLUSIONES

En este trabajo se ha desarrollado un sistema de detección de anomalías en el tráfico de red, como continuación de la labor de Benjamín Martín en su estudio sobre la predictibilidad del tráfico [3]. Como estudiante del Doble Grado en Ingeniería Informática y Matemáticas, he podido aprovechar los conocimientos adquiridos durante la carrera para abordar el análisis de las series temporales en el contexto del sistema de detección de anomalías en el tráfico de red. Particularmente, el conocimiento adquirido en las asignaturas de Redes de Comunicaciones y Estadística han sido de gran utilidad para el desarrollo de este trabajo.

Con el fin de mejorar el sistema inicial propuesto por B. Martín, se ha obtenido una serie estacionaria a partir de métodos de diferenciación discreta y convoluciones. Posteriormente, se ha aplicado una metodología basada en la estandarización de los datos, que nos ha permitido comparar los datos de las series temporales. En concreto, se ha estudiado la distribución de los parámetros de la distribución α -estable de la serie diferenciada y también su tipificación, bajo la hipótesis de que la serie diferenciada sigue un proceso estacionario. Se ha encontrado que el parámetro *mean* de los datos tipificados es un buen indicador para la detección de anomalías, y que la combinación del parámetro *mean* y δ de la distribución α -estable puede mejorar la detección de anomalías. Además, al emplear todos los parámetros combinados (los cuatro parámetros de ajuste α -estable: α , β , γ y δ ; y los dos parámetros obtenidos mediante la tipificación: *mean* y *std*) en una regresión logística, se observa una ligera mejora de las predicciones de anomalías, reduciendo tanto los falsos positivos como los falsos negativos.

Cabe destacar que en este trabajo se aplican bases matemáticas sólidas, como la diferenciación discreta y mediante convoluciones, que nos proporciona una serie estacionaria. Se han utilizado propiedades de dicha serie estacionaria para poder comparar los datos de las series temporales y detectar anomalías.

Se ha concluido que la detección de anomalías es adecuada, aunque la cantidad de falsos positivos y falsos negativos puede ser elevada para un centro de operaciones de ciberseguridad. Sin embargo, debido a la simplicidad del proceso y poco coste computacional, se considera que es un buen sistema de detección temprana de anomalías.

6.1. Trabajos futuros

A pesar de las ventajas significativas del sistema propuesto para la detección de anomalías en el tráfico, como su enfoque genérico y la simplicidad en la predicción de anomalías, existen diversas oportunidades para continuar desarrollándolo y refinándolo. Estas mejoras pueden abordar desafíos específicos y aumentar la efectividad del sistema en entornos de ciberseguridad más complejos y dinámicos. A continuación se detallan algunas posibles mejoras:

- **Ajustar la sensibilidad de la detección:** Actualmente, el número de falsos positivos y falsos negativos aplicando la regresión logística sobre todos los parámetros es similar, entre 15 % y 20 %. Aunque el sistema es adecuado para detectar anomalías, puede ser de interés ajustar la sensibilidad de la detección para reducir el número de falsos positivos y falsos negativos.
- **Mejorar la detección de anomalías:** Se ha encontrado que la combinación de los parámetros del ajuste α -estable y los parámetros obtenidos por la tipificación de los datos es eficaz para la detección de anomalías. Sin embargo, se puede investigar el uso de predicciones previas para mejorar la detección de anomalías. Como se ha visto en el capítulo 5, en el apartado 5.4.1, el número de falsos positivos no parecían concentrarse en algunos rangos, al contrario que los falsos negativos. Por tanto, se puede investigar sobre un método de predicción que tenga en cuenta múltiples ventanas temporales adyacentes para mejorar la detección de anomalías.
- **Búsqueda de otros parámetros:** En este trabajo se han utilizado los parámetros de la distribución α -estable y los parámetros *mean* y *std* obtenidos mediante la tipificación de los datos. Sin embargo, se pueden investigar otros parámetros que puedan mejorar la detección de anomalías. Por ejemplo, se plantea investigar el uso de los cuantiles de la serie diferenciada, con el fin de buscar valores extremos que puedan indicar la presencia de anomalías.

BIBLIOGRAFÍA

- [1] G. Maciá-Fernández, J. Camacho, R. Magán-Carrión, P. García-Teodoro, and R. Therón, “UGR’16: A new dataset for the evaluation of cyclostationarity-based network IDSs,” *Computers & Security*, vol. 73, pp. 411–424, 2018.
- [2] E. Cabornero Pinto, “Análisis de tráfico de internet mediante el uso del suavizado exponencial de series temporales,” *Universidad Autónoma de Madrid*, 2021.
- [3] B. Martín Gómez, “Estudio de la predictibilidad del tráfico en internet para la detección de anomalías sutiles,” *Universidad Autónoma de Madrid*, 2023.
- [4] B. Martín Gómez, “Detección de ciberataques de denegación de servicio mediante funciones características,” *Universidad Autónoma de Madrid*, 2021.
- [5] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, “Developing realistic distributed denial of service (ddos) attack dataset and taxonomy,” in *2019 International Carnahan Conference on Security Technology (ICCST)*, pp. 1–8, 2019.
- [6] I. Strelkovskaya, I. Solovskaya, and A. Makoganiuk, “Spline-extrapolation method in traffic forecasting in 5G networks,” *Journal of Telecommunications and Information Technology*, no. 3, pp. 8–16, 2019.
- [7] Y. Zhang, Z. M. Mao, and J. Wang, “Low-rate TCP-targeted dos attack disrupts internet routing.,” in *Proc. 14th Annual Network and Distributed System Security (NDSS)*, (San Diego, CA, USA), Internet Society, Feb. 2007.
- [8] K. S. Bhosale, M. Nenova, and G. Iliev, “The distributed denial of service attacks (ddos) prevention mechanisms on application layer,” in *2017 13th International Conference on Advanced Technologies, Systems and Services in Telecommunications (TELSIKS)*, pp. 136–139, IEEE, 2017.
- [9] J. P. Nolan, “Univariate stable distributions,” *Springer Series in Operations Research and Financial Engineering*, vol. 10, pp. 978–3, 2020.
- [10] M. Stoppa, “Estimación y comparación de un modelo estadístico α -estable de primer orden basado en flujos NetFlow respecto al tráfico agregado en redes IP,” *Universidad Autónoma de Madrid*, 2013.
- [11] F. Simmross-Wattenberg, J. I. Asensio-Perez, P. Casaseca-De-La-Higuera, M. Martin-Fernandez, I. A. Dimitriadis, and C. Alberola-Lopez, “Anomaly detection in network traffic based on statistical inference and α -stable modeling,” *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 4, pp. 494–509, 2011.
- [12] H. Prautzsch, W. Boehm, and M. Paluszny, *Bézier and B-spline techniques*, vol. 6. Springer, 2002.
- [13] D. Muelas Recuenco, “Aplicación de análisis funcional de datos a la gestión de redes,” *Universidad Autónoma de Madrid*, 2015.

- [14] C. Chatfield, "The holt-winters forecasting procedure," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 27, no. 3, pp. 264–279, 1978.
- [15] S. Kirkpatrick, C. Gelatt Jr, and M. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [16] D. A. Dickey and W. A. Fuller, "Distribution of the estimators for autoregressive time series with a unit root," *Journal of the American Statistical Association*, vol. 74, no. 366, pp. 427–431, 1979.
- [17] D. Kwiatkowski, P. C. B. Phillips, P. Schmidt, and Y. Shin, "Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root?," *Journal of Econometrics*, vol. 54, no. 1-3, pp. 159–178, 1992.
- [18] A. N. Kolmogorov, "Sulla determinazione empirica di una legge di distribuzione," *Giornale dell'Istituto Italiano degli Attuari*, vol. 4, pp. 83–91, 1933.
- [19] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.
- [20] D. A. Salazar, J. I. Vélez, and J. C. Salazar, "Comparison between SVM and logistic regression: Which one is better to discriminate?," *Revista Colombiana de Estadística*, vol. 35, no. 2, pp. 223–237, 2012.

ACRÓNIMOS

ADF Dickey-Fuller Aumentado, *Augmented Dickey-fuller*.

CIC Instituto de Ciberseguridad de Canadá, *Canadian Institute for Cybersecurity*.

DDoS Denegación de Servicio Distribuidos, *Distributed Denial of Service*.

DoS Denegación de Servicio, *Denial of Service*.

FDA Análisis de Datos Funcionales, *Functional Data Analysis*.

GLRT Prueba de Razón de Verosimilitud Generalizada, *Generalized Likelihood Ratio Test*.

HTTP Protocolo de Transferencia de Hipertexto, *Hypertext Transfer Protocol*.

ICMP Protocolo de Mensajes de Control de Internet, *Internet Control Message Protocol*.

KPSS Kwiatkowski-Phillips-Schmidt-Shin.

K-S Kolmogorov-Smirnov.

MSE Error Cuadrático Medio, *Mean Squared Error*.

RBF Función de Base Radial, *Radial Basis Function*.

RNN Redes Neuronales Recurrentes, *Recurrent Neural Networks*.

ROC Característica Operativa del Receptor, *Receiving Operating Characteristic*.

SVM Máquinas de Vectores de Soporte, *Support Vector Machine*.

TFG Trabajo de Fin de Grado.

TFM Trabajo de Fin de Máster.

UDP Protocolo de Datagramas de Usuario, *User Datagram Protocol*.

UGR Universidad de Granada.

UNB Universidad de New Brunswick.

APÉNDICES

ENLACE A CÓDIGO

El código que se ha hecho durante el desarrollo de este trabajo se encuentra en el siguiente enlace para facilitar trabajos futuros. Se ha utilizado el lenguaje de programación Python y las librerías de análisis de datos Pandas, Numpy y Scipy. El código se ha organizado en diferentes scripts para facilitar su comprensión y reutilización. En el siguiente enlace se puede acceder a los scripts de Python utilizados en este trabajo: <https://github.com/Bubbasm/tfginfo>



Universidad Autónoma
de Madrid