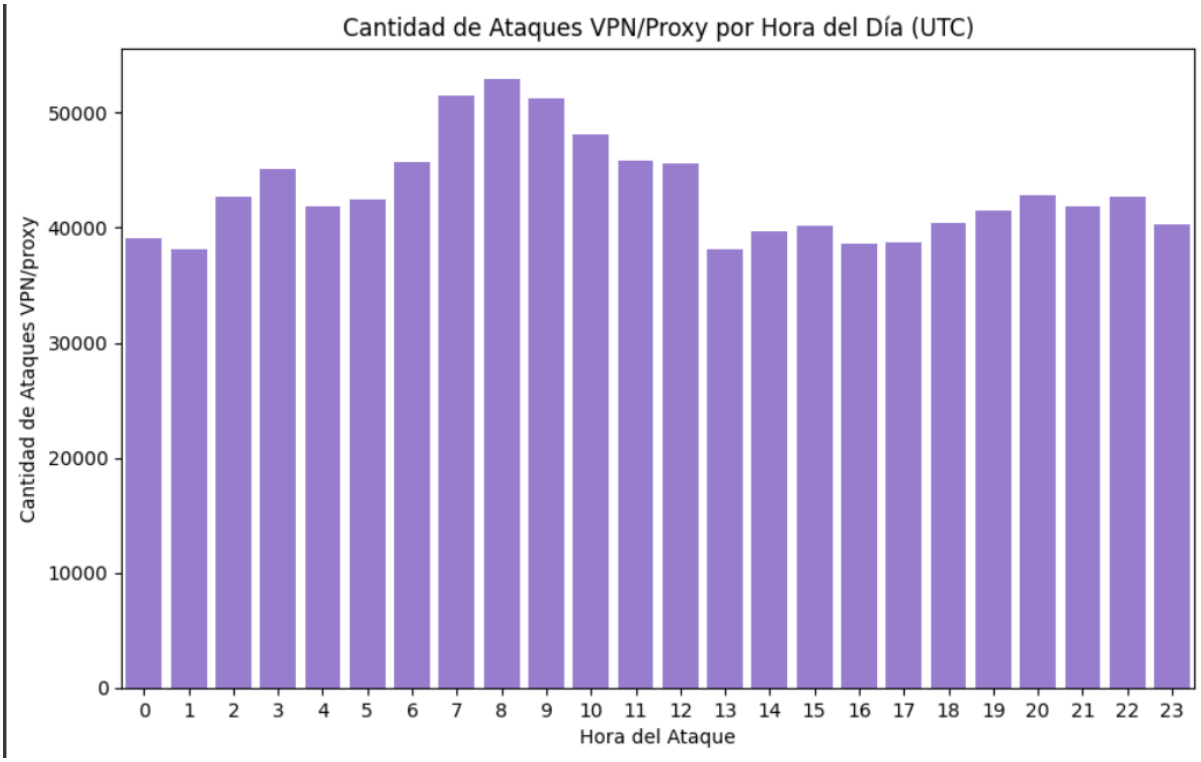
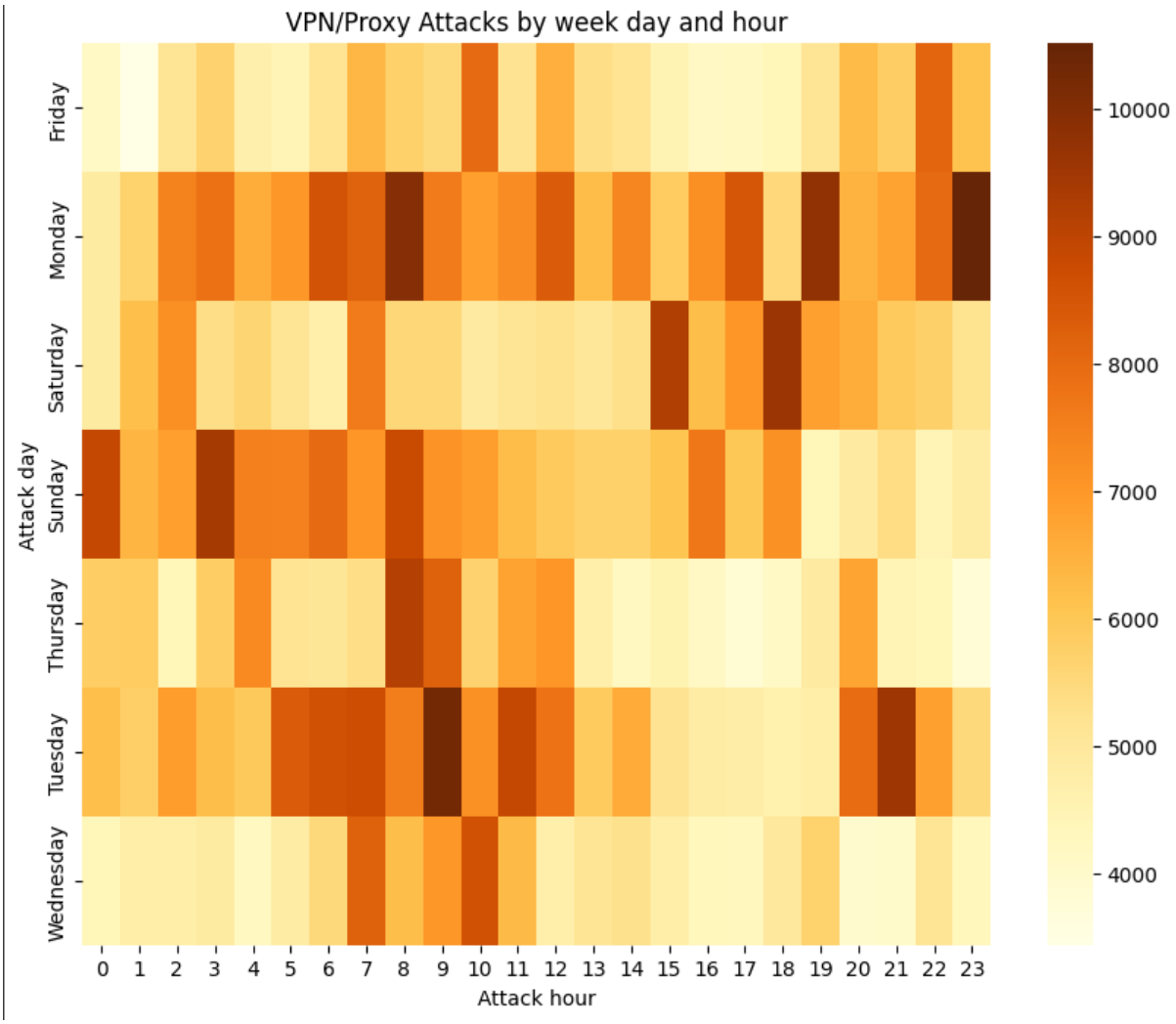


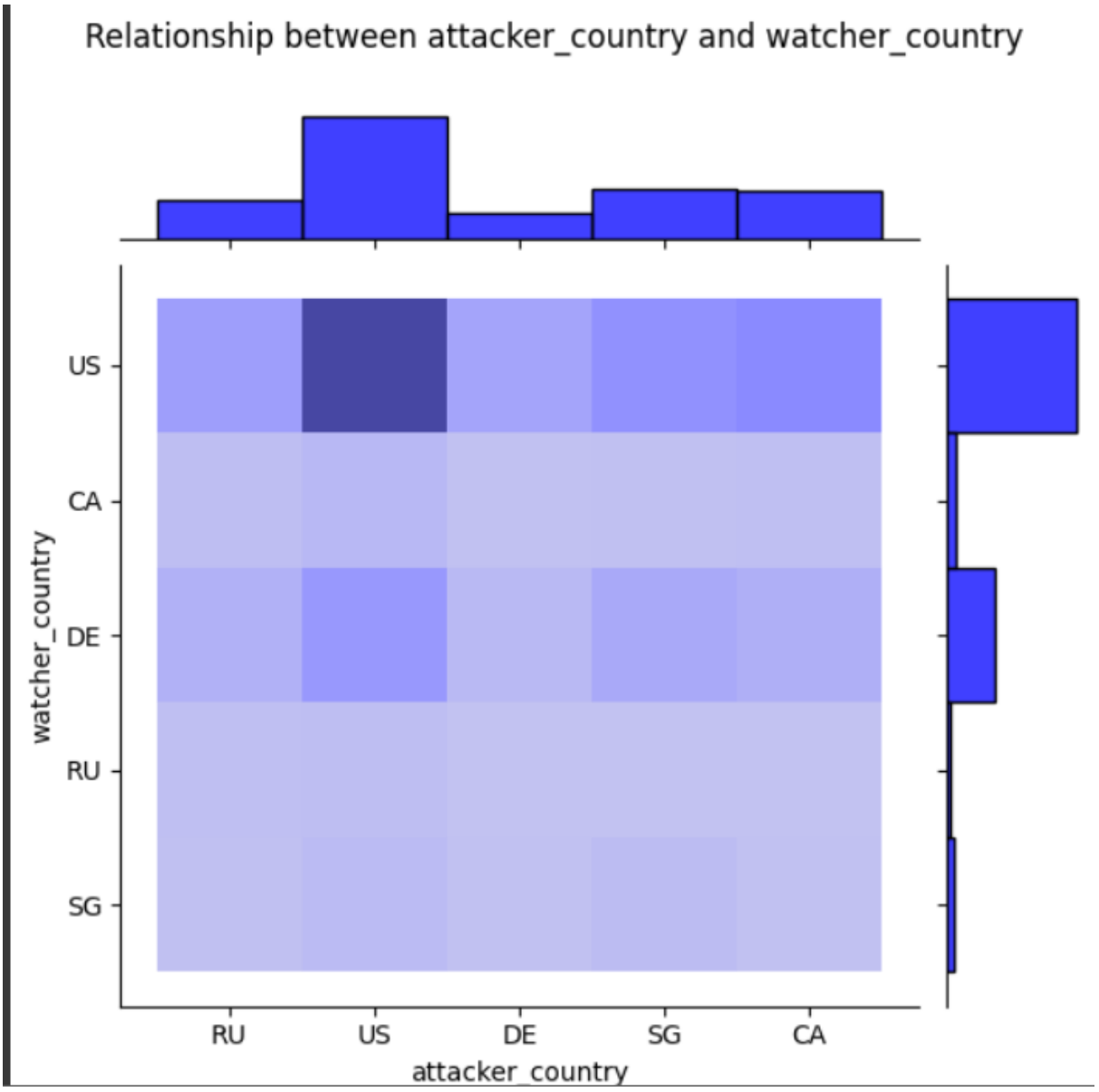
Link a colab: [barplot]

Este plot lo realice con el fin de ver si existía alguna relación entre el horario del ataque y que este sea VPN o Proxy. Quería ver si en algún horario había una gran diferencia en cuanto a cantidad de ataques de este tipo para extraer alguna feature interesante, aunque no me resultó útil para los modelos.



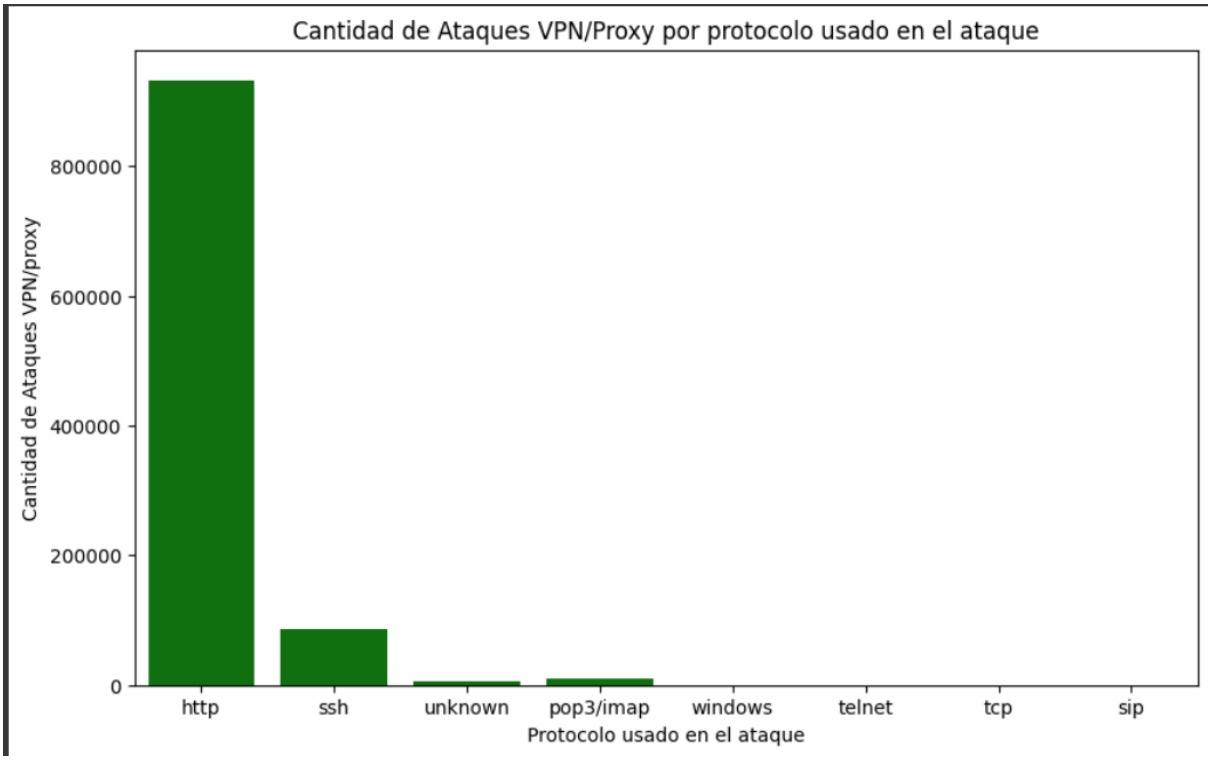
Link a colab: heatmap

En este plot también quería ver la relación del tiempo del ataque con que fuera VPN o Proxy. Me sirvió que en ciertas franjas horarias y ciertos días específicos de la semana se frecuentan más.



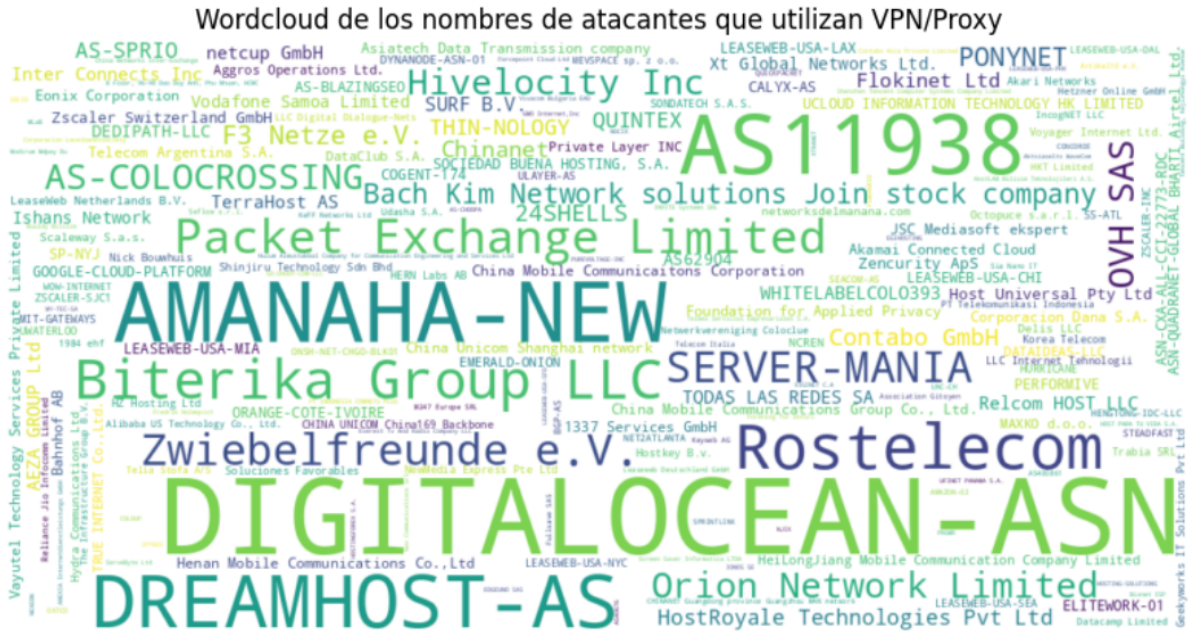
Link a colab: [plot2D]

En este plot decidi incluir unicamente a los 5 paises que mas se repetian en “attacker_country” y “watcher_country” de los ataques VPN para ver la distribucion de los paises que mas eran atacados y los que mas atacaban con VPN.



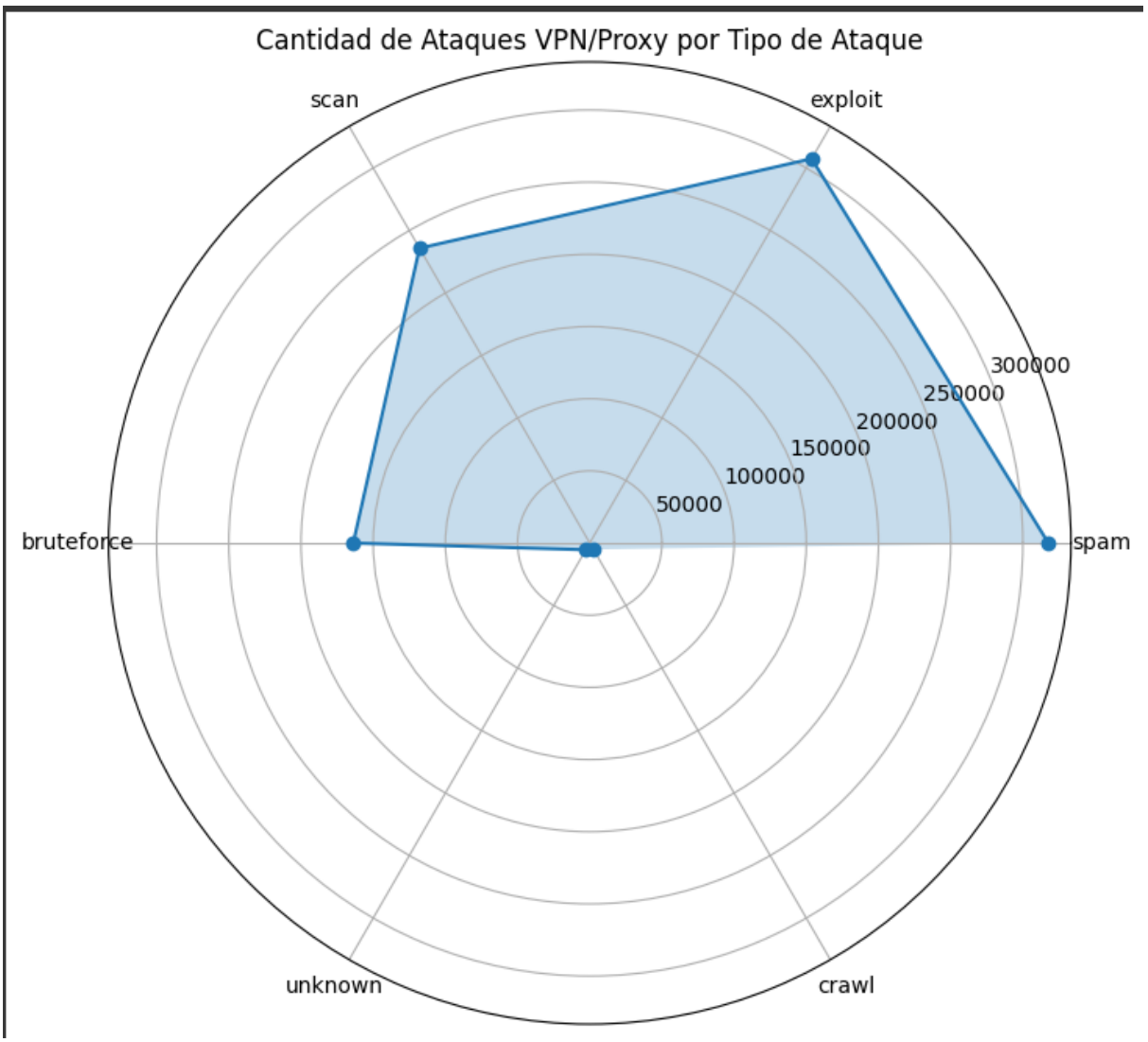
Link a colab: [obligatoriaIV]

Este plot fue importante para definir cuál era el servicio más utilizado en los ataque VPN o proxy. Aporto una alta relación con el label y sirvió para crear nuevas features.



Link a colab: [obligatoriaV]

Este wordcloud sirvió para ver cómo se distribuye la cantidad de ataque por “attacker_as_name” y ayudó a visualizar las más importantes.

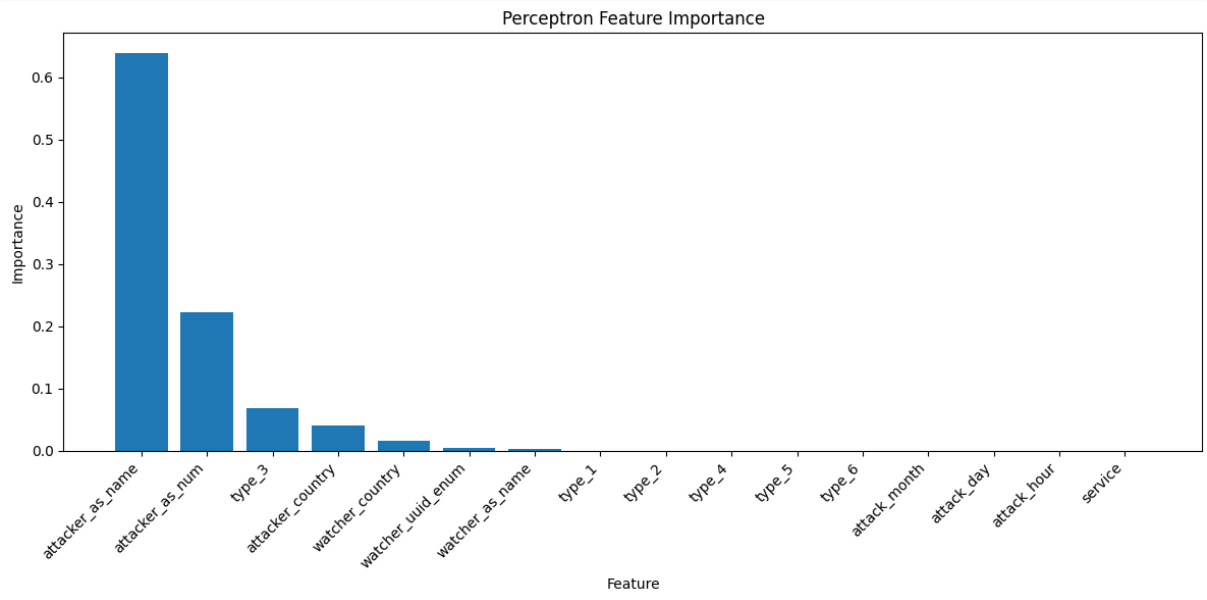


Link a colab: [obligatoriaV]

Este plot definió la distribución de los ataques VPN o Proxy de acuerdo al tipo de ataque realizado y ayudó en la creación de una nueva feature también.

Baseline

Link al colab del Baseline



[Link a colab con plot F1]

El feature Importance sobre el modelo más básico de todos, aporto una idea de cuales eran las features a prestar atención para un feature engineering óptimo.

- F1- Val: **0.649126816093125**
- F1- Test: **0.50297**
- Features:
    - feature 1: Service (encodeada como Ordinal y normalizada) : Extraer de la columna “attack_type” el servicio (http, ssh, windows, etc)
    - feature 2: Type (type_1, …, type_6 encodeada como ohe): Extraer de la columna “attack_type” únicamente el tipo de ataque (bruteforce, spam, scan, force, etc)
    - feature 3: Attack_month (encodeada como Ordinal y normalizada)
    - feature 4: Attack_day (encodeada como Ordinal y normalizada)
    - feature 4: Attack_hour (encodeada como Ordinal y normalizada)
    - feature 6: watcher_as_name (encodedada como Ordinal y normalizada)
    - feature 7: watcher_uuid_enum (normalizada)
    - feature 8: watcher_country (encodeada como ordinal y normalizada)
    - feature 9: attacker_country (encodeada como ordinal y normalizada)
    - feature 10: attacker_as_num (normalizada)
    - feature 11: attacker_as_name (encodeada con mean encoding)

Mejor Modelo

Link al colab del mejor modelo

- Elegí hacer este modelo porque: Este modelo es útil para resolver relaciones no lineales y esa fue una de las principales razones por las que lo elegí. Este problema no me pareció para nada lineal y sobre todo fue muy difícil poder ver relaciones claras entre los datos. Además sabía que me iba a funcionar bien para los datos que ya tenía y había utilizado para el baseline.
- Este es el mejor modelo porque: Al ser una red neuronal que sirve para modelar datos y relaciones no lineales, es capaz de generalizar bien problemas complejos como este a pesar de no tener relaciones evidentes entre los datos.
- F1 - Val: **0.7177400664383387**
- F1- Test: **0.58939**
- Features:
    - Feature 1: *Service* : Extraer de la columna “attack_type” el servicio (http, ssh, windows, etc)
    - Feature 2: *Type*: Extraer de la columna “attack_type” únicamente el tipo de ataque (bruteforce, spam, scan, force, etc)
    - Feature 3: *Number_of_open_ports* : Por cada IP en el archivo shodan, la cantidad de puertos abiertos para esa IP.
    - Feature 4: Ref_ports : Por cada Ip en el archivo shodan, me fijaba si algun puerto indicativo de VPN o Proxy estaba abierto. Para eso me base en esto (https://www.speedguide.net/port.php?port=8080) (https://nordvpn.com/es/blog/what-are-vpn-ports/#:~:text=The%20most%20common%20VPN%20ports,IKEv2%2C%20and%201723%20for%20PPTP.)
    - Feature 5: Foreign_attacker_country : Un 1 si el attacker_country y el watcher_country son distintos o un 0 en caso contrario
    - feature 6: Attack_month (encodeada como Ordinal y normalizada)
    - feature 7: Attack_day (encodeada como Ordinal y normalizada)
    - feature 8: Attack_hour (encodeada como Ordinal y normalizada)
    - feature 9: watcher_as_name (encodedada como Ordinal y normalizada)
    - feature 10: watcher_uuid_enum (normalizada)
    - feature 11: watcher_country (encodeada como ordinal y normalizada)
    - feature 12: attacker_country (encodeada como ordinal y normalizada)
    - feature 13: attacker_as_num (normalizada)
    - feature 14: attacker_as_name (encodeada con mean encoding)

Segundo Modelo

Link al colab del segundo modelo

- Elegí hacer este modelo porque: Una de las principales razones es que estos modelos tienen la capacidad de entrenar muy rápido para conjuntos grandes de datos, además de que cuentan con el método “partial_fit” el cual resulta muy util para que vaya aprendiendo de manera incremental. Ademas es eficiente para manejar muchas features y puede manejar bien el problema del desbalanceo. Me base en este articulo para elegirlo, dado que para este problema era importante la eficiencia en tiempo (una de las principales razones por las que decidí elegirlo)
- F1 - Val: **0.6530211522020903**
- F1- Test: **0.51606**
- Features:
    - Feature 1: *Service* : Extraer de la columna “attack_type” el servicio (http, ssh, windows, etc)
    - Feature 2: *Type*: Extraer de la columna “attack_type” únicamente el tipo de ataque (bruteforce, spam, scan, force, etc)
    - Feature 3: *Number_of_open_ports* : Por cada IP en el archivo shodan, la cantidad de puertos abiertos para esa IP.
    - Feature 4: Ref_ports : Por cada Ip en el archivo shodan, me fijaba si algun puerto indicativo de VPN o Proxy estaba abierto. Para eso me base en esto (https://www.speedguide.net/port.php?port=8080) (https://nordvpn.com/es/blog/what-are-vpn-ports/#:~:text=The%20most%20common%20VPN%20ports,IKEv2%2C%20and%201723%20for%20PPTP.)
    - Feature 5: Foreign_attacker_country : Un 1 si el attacker_country y el watcher_country son distintos o un 0 en caso contrario
    - feature 6: Attack_month (encodeada como Ordinal y normalizada)
    - feature 7: Attack_day (encodeada como Ordinal y normalizada)
    - feature 8: Attack_hour (encodeada como Ordinal y normalizada)
    - feature 9: watcher_as_name (encodedada como Ordinal y normalizada)
    - feature 10: watcher_uuid_enum (normalizada)
    - feature 11: watcher_country (encodeada como ordinal y normalizada)
    - feature 12: attacker_country (encodeada como ordinal y normalizada)
    - feature 13: attacker_as_num (normalizada)
    - feature 14: attacker_as_name (encodeada con mean encoding)

Link a carpeta con todos los archivos, csv de predicciones y visus