Aprendizaje Automático

Modelos de aprendizaje no supervisados

Integrantes:

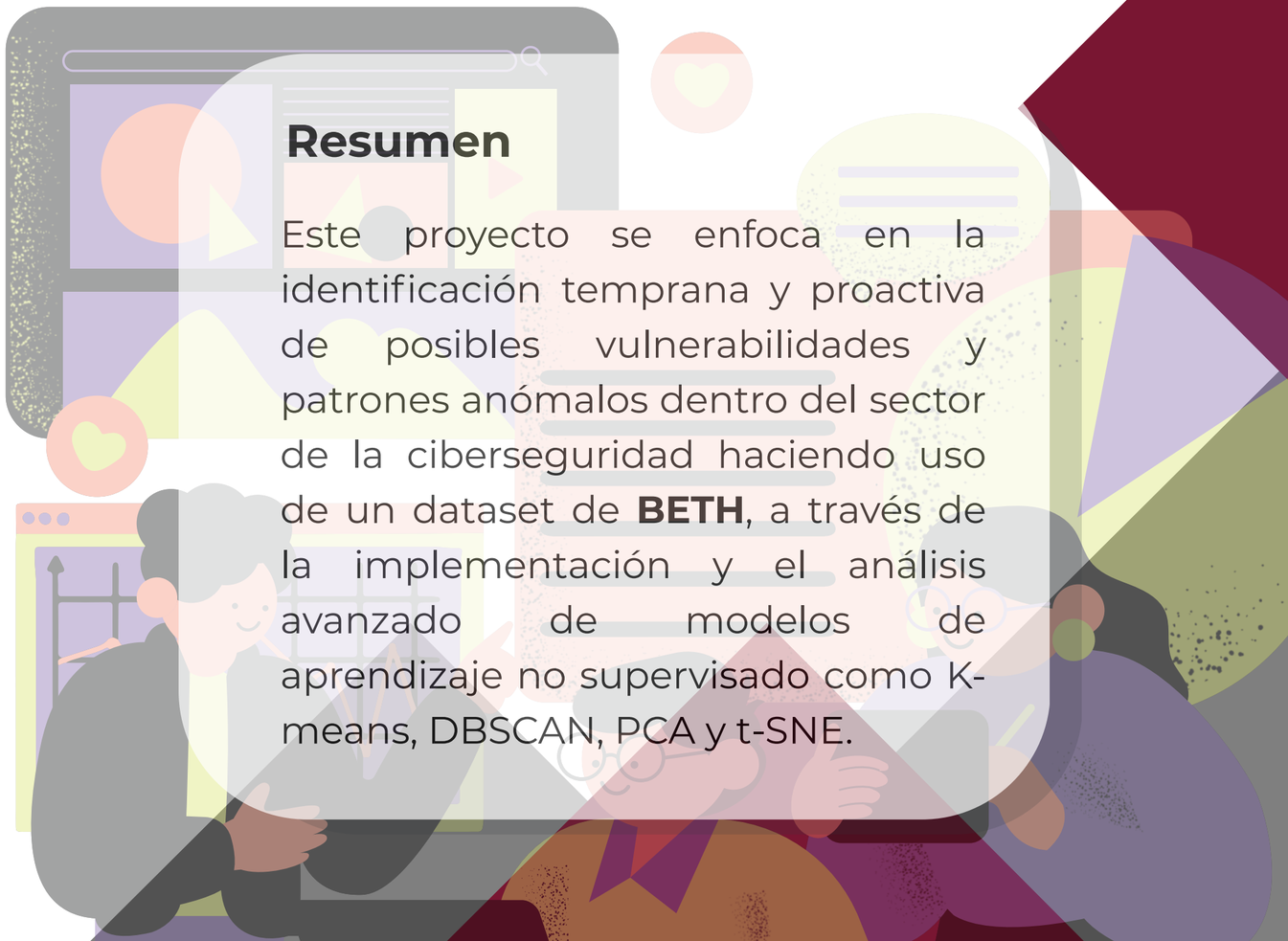
- Bryan Ismael Alvarado Quimiz
- Diego André Arámbulo Vítores
- Jesús Antonio Zambrano Parrales

Entregable 1: Repositorio

https://github.com/diegoarambulo/2025_301_MIAR0525_A01_SEM2_GRUPAL/tree/main

Entregable 2: Video presentación

https://youtu.be/lWwAwwNrHdU



Resumen

Este proyecto se enfoca en la identificación temprana y proactiva de posibles vulnerabilidades y patrones anómalos dentro del sector de la ciberseguridad haciendo uso de un dataset de **BETH**, a través de la implementación y el análisis avanzado de modelos de aprendizaje no supervisado como K-means, DBSCAN, PCA y t-SNE.

Introducción del problema

La detección de amenazas desconocidas y la identificación proactiva de vulnerabilidades en entornos de ciberseguridad requieren de metodologías que superen las limitaciones de los datos etiquetados. Se establece que el reto es superar las "limitaciones de los datos etiquetados", una debilidad común en la ciberseguridad, justificando el uso del aprendizaje no supervisado, que en conjunto con el uso de técnicas de creación de clusters, podremos segmentar de forma mas adecuada una amenaza detectada.

Metodología y técnicas aplicadas

Primero cargamos Beth v3, comprobamos tipos y valores ausentes y nos quedamos con variables numéricas clave: hora convertida a **'hour_sin','hour_cos','processNameConverted','eventId','evil'**. Todas se normalizaron con **Standard Scaler** para que cada componente aportara por igual a la distancia euclidiana. Con esa matriz aplicamos **K-Means, variando k de 2 a 8** y eligiendo el máximo índice de codo; se formaron tres grupos bien diferenciados de tráfico diario, sospechoso y fuertemente malicioso.

Después ejecutamos DBSCAN explorando radios (eps) entre 0,05 y 3 y mínimos de vecinos de 3 a 12; seleccionamos la combinación que generó dos clústeres densos + ruido y ofreció la mejor silueta. Para interpretar, comprimimos con PCA y t-SNE: en ambos planos el clúster rojo concentró la mayoría de eventos evil, mientras el ruido capturó llamadas con retornos de error extremos, confirmando visualmente la eficacia del agrupamiento sin utilizar codificación categórica.

Análisis comparativo entre modelos

¿Qué tipo de perfiles se pueden identificar?

- Entre los perfiles mas notables podemos identificar los siguientes:
  - Ataques cibernéticos realizados en horarios diurnos
  - Ataque por apertura de archivos
  - Ataque iniciados por procesos de sistema
  - Ataque por apertura de conexión

¿Qué diferencias clave surgieron entre los modelos?

- K-means, al utilizar como base la distancia euclidiana, revelara clusters limpios y esféricos, podemos considerarlo como la base.
- DBSCAN, en cambio revelara cluster arbitrarios etiquetando los outliers con -1, ya que se basa en densidad mas no en distancia, útil cuando no hay balance en las clases, pero agrega ruido.
- PCA, siendo realmente mas alineado a una reducción de dimensionalidad, se refleja de forma global y preserva la varianza, no tiene captación para estructuras no lineales, mayormente usado para identificar direcciones mas no separar cluster
- T-SNE, al igual que PCA sirve mas para reducir densidad, para data no lineal. Revela cluster bien definidos, clusters de puntos cercanos y compactos, ademas de la posibilidad de mostrar continuidades escondidas

Análisis comparativo entre modelos

¿Qué limitaciones encontraron y cómo las abordarían?

- **Implementación K-means**
  - se asume cluster esféricos y altamente definidos **===>** limpieza de outliers antes de clusterizar
  - alta sensibilidad con los outliers **===>** si la forma del cluster no es esferiza, implementar K-medoids o modelo gauseano de mezcla
- **Implementación DBSCAN**
  - hiperparámetros difíciles de pulir **===>** para el ajuste eps, se podría usar k-dist plot en conjunto con grid
  - baja eficiencia en datasets de grande dimension **===>** para un manejo de densidades variadas, se usaría *High density BSCAN*
- **Implementación PCA**
  - pierde patrones complejos al capturar solo relaciones lineales **===>** realizar una selección de features sumado a una normalización antes de usar el PCA
  - existe posibilidad de mezclar ruido y señal **===>** mezclarlo con métodos no lineales para capturas mas finas
- **Implementación t-SNE**
  - genera clusters atractivos que pueden ser falsos **===>** usarlo solo de forma exploratoria
  - alta sensibilidad a hiperparámetros siendo inestable **===>** Ajustar learning rate y validar con otros métodos

Conclusiones y recomendaciones.

Con los resultados obtenidos después de aplicar los diferentes modelos de clusterización aplicando ML no etiquetado y reducción de dimensionalidad, podemos llegar a las siguientes conclusiones:

- Con la capacidad de sectorizar los puntos de impacto, tal cual nos indica los clusters, podremos ser mas eficientes y estratégicos al momento de ejecutar un mecanismo de contingencia.
- Al momento de realizar las transformaciones necesarias, debemos tener en cuenta que usar datos lineales le darán mayor sentido a nuestra almacen de informacion y por consiguiente en todos los dashboards que usemos. (recordemos que en el caso de la base *BETH* no existen un numero natural de clusters por lo que la definición de K para K-means puede tornarse conflictiva)
- No existe un modelo de ML perfecto para todos los problemas cotidianos, un analisis y comparativa es necesario para tomar el mas eficiente y adecuado a nuestras necesidades, métodos de redimensionamiento como el PCA pueden ser difíciles de interpretar debido a su capacidad de mezclar señales con ruidos, no deberían ser definitivos en estructuras finales
- El uso de t-sne, es correcto si solo haremos exploración visual de los resultados, ya que estos serán de carácter ficticio, y no deben usarse para segmentación.