Entrega 2

Responsables:

Daniel Quintero Álvarez

Profesor:

Raul Ramos Pollan

Universidad de Antioquia

Introducción a la inteligencia artificial

2023/2

1. Planteamiento del problema

El tema que se está tratando es cómo identificar y categorizar páginas que hacen phishing, que es una forma de estafa por internet. El anti-phishing son las acciones que se toman para evitar los ataques de phishing. El phishing es un crimen informático en el que los estafadores se hacen pasar por entidades confiables o conocidas y se comunican con las personas por diferentes medios, como email, mensajes de texto o teléfono, para obtener información privada. Normalmente, en un ataque de phishing por email, el mensaje falso dirá que hay un problema con una factura, que ha habido actividad sospechosa en una cuenta o que el usuario debe iniciar sesión para verificar una cuenta o contraseña. Además, los estafadores pueden pedir a los usuarios que ingresen información de la tarjeta de crédito, detalles bancarios y otros datos personales sensibles. Una vez que los estafadores recogen esta información, pueden usarla para acceder a cuentas, robar datos e identidades, así como descargar malware en la computadora del usuario. Por eso, es importante tener medidas de seguridad efectivas para prevenir los ataques de phishing y proteger la privacidad y seguridad en línea de los usuarios.

1.2.Dataset

El dataset seleccionado es Phishing website dataset (https://www.kaggle.com/datasets/akashkr/phishing-website-dataset/data) Este conjunto de datos contiene 30 atributos extraídos de 11000 páginas.

Algunos de los atributos más significativos son:

- Iframe contains: [1 -1]
- age_of_domain contains: [-1 1]
- DNSRecord contains: [-1 1]
- web_traffic contains: [-1 0 1]
- Page_Rank contains: [-1 1]
- Google_Index contains: [1 -1]
- Links_pointing_to_page contains: [1 0 -1]
- Statistical_report contains: [-1 1]
- Result contains [-1 1]

Según la descripción de los datos, estos son el significado de los valores en los datos.

1 significa legítimo

0 es sospechoso

-1 es phishing

1.3. Métricas

Para medir el desempeño del sistema se usarán dos medidas de evaluación principales: el accuracy y el f1 score, ya que ambos se centran en la exactitud. Además de estas medidas técnicas, se considera la medida de negocio, la confianza de las predicciones para saber si una página tiene phishing o no. Es esencial que estas predicciones sean confiables para que el navegador web pueda impedir que sus usuarios entren a páginas dañinas.

1.4. Desempeño

En un modelo de este tipo, se busca que la exactitud de las predicciones sea alta, superando el 80%, además será importante evitar un gran número de falsos positivos. En un ambiente productivo, el modelo sería usado como un filtro para evitar que los usuarios entren a páginas sospechosas y, así, asegurar la seguridad de los usuarios. Por lo tanto, es clave que el modelo pueda ofrecer predicciones confiables y precisas para cumplir este objetivo.

2. Exploración descriptiva del dataset

La base de datos utilizada en este proyecto es el "Phishing website dataset" y consta de 11,055 muestras y 32 columnas. De estas variables, 31 se utilizan para describir las características de una URL de un sitio web, mientras que la otra variable es la salida "Result", que indica si el sitio web es o no un sitio de phishing.

La información contenida en las 30 variables descriptivas se utiliza para identificar patrones y características comunes entre los sitios web de phishing. Al analizar estas variables, se pueden crear modelos de aprendizaje automático que puedan detectar de manera efectiva la presencia de sitios web de phishing y proteger a los usuarios contra posibles ataques.

El nombre de las columnas presentes en el dataset son las siguientes:

```
    index
0   index
1   having_IPhaving_IP_Address
2   URLURL_Length
3   Shortining_Service
4   having_At_Symbol
5   double_slash_redirecting
6   Prefix_Suffix
7   having_Sub_Domain
8   SSLfinal_State
9   Domain_registeration_length
10  Favicon
11  port
12  HTTPS_token
13  Request_URL
14  URL_of_Anchor
15  Links_in_tags
16  SFH
17  Submitting_to_email
18  Abnormal_URL
19  Redirect
20  on_mouseover
21  RightClick
22  popUpWidnow
23  Iframe
24  age_of_domain
25  DNSRecord
26  web_traffic
27  Page_Rank
28  Google_Index
29  Links_pointing_to_page
30  Statistical_report
31  Result
```

Además, se decidió quitar la columna "INDEX" durante el análisis, ya que esta variable no es importante para el objetivo del estudio.

La columna "INDEX" suele ser un identificador único asignado a cada muestra en un conjunto de datos. En la mayoría de los casos, esta variable no tiene ninguna influencia en la predicción o en el análisis de los datos. Es solo un número de referencia que se usa para identificar la muestra en la base de datos.

Por lo tanto, quitar la columna "INDEX" no afectará la calidad de los resultados y puede facilitar el análisis, ya que se disminuirá la cantidad de variables en el conjunto de datos. Esto también puede ayudar a mejorar el rendimiento y la eficiencia de los modelos de aprendizaje automático, ya que tendrán menos variables que analizar. En general, la eliminación de variables irrelevantes puede ser una práctica habitual en el

preprocesamiento de datos para mejorar la precisión y la eficiencia en el análisis de los datos.

Elimación de la columna Index

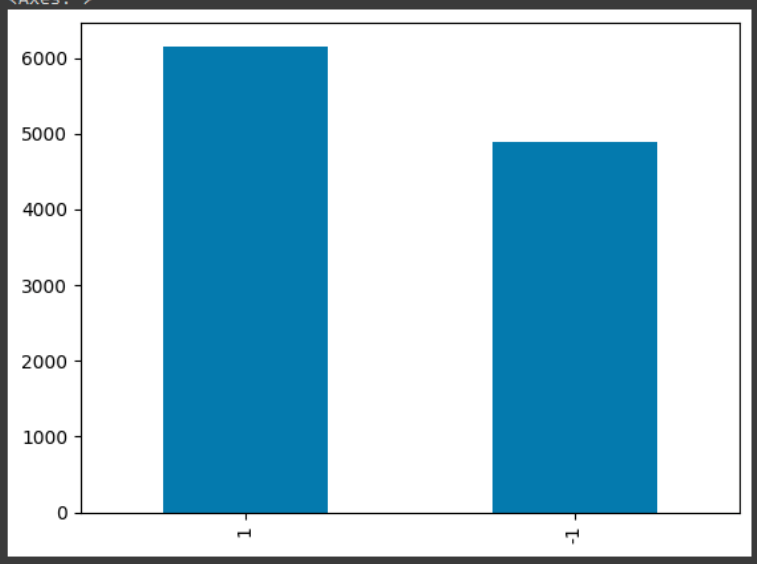
```
[ ]  del data["Index"]
```

2.1 Balanceo de los datos

Después de explorar el conjunto de datos, se observó que no hay desequilibrio en los datos. El dataset consta de un total de 10000 muestras, de las cuales 6157 pertenecen a la clase 1 (no phishing) y 4898 pertenecen a la clase -1 (phishing).

Balanceo de Result

```
[14] data['Result'].value_counts().plot(kind='bar')
```



Según el análisis de balanceo, se puede decir que el conjunto de datos tiene una distribución bastante equilibrada entre las dos clases. La clase 1 (no phishing) tiene un 61.57% de las muestras, mientras que la clase -1 (phishing) tiene un 48.98%. Esto significa que no hay una clase dominante o minoritaria que pueda sesgar los resultados del modelo. Un conjunto de datos equilibrado es deseable para que el modelo pueda aprender a distinguir correctamente entre las dos clases y no se vea afectado por el desbalanceo.

Se dividió él dataset en dos bloques destinados al entrenamiento de los datos y validación del modelo.

Observamos la mejor accuracy con profundidad del árbol 10.

```
[ ] resultados_dt = experimentar_dt([3,10,50,100],MinMaxScaler().fit_transform(X), Y)
    resultados_dt
```

	profundidad del arbol	eficiencia de entrenamiento	desviacion estandar entrenamiento	eficiencia de prueba	desviacion estandar prueba	accuracy
0	3.0	0.908227	0.001163	0.904029	0.011213	0.904029
1	10.0	0.958571	0.001633	0.944277	0.006502	0.944277
2	50.0	0.989889	0.000318	0.961458	0.020049	0.961458
3	100.0	0.989889	0.000318	0.962091	0.019774	0.962091

Se están realizando pruebas eliminando datos random del dataset

```
remove_n = 3000
drop_indices = np.random.choice(data.index, remove_n, replace=False)
df_subset = data.drop(drop_indices)

X = df_subset.drop('Result', axis=1).values
Y = df_subset['Result'].values
print (X.shape , Y.shape)
```

Referencias

https://www.kaggle.com/datasets/akashkr/phishing-website-dataset/data

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html

https://support.minitab.com/es-mx/minitab/21/help-and-how-to/statistical-modeling/regression/supporting-topics/basics/what-are-categorical-discrete-and-continuous-variables/

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html

https://scikit-learn.org/stable/modules/tree.html#tree