Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2115 – Programación como Herramienta para la Ingeniería (I/2018)

Actividad 12

Objetivos

- Utilizar diferentes herramientas de análisis de datos en Python.
- Generar predicciones utilizando un modelo de árbol de decisión.

Entrega

- **Lenguaje a utilizar:** Python
- **Lugar:** GitHub
- **Hora:** 17:30

Introducción

Bastián ha estado metido en movidas un tanto "truchas" últimamente. Se rumorea que ha estado involucrado en el tráfico de órganos. Revisamos el historial de su navegador y encontramos cosas horribles. Entre ellas una serie de páginas sospechosas que podrían estar destinadas a hacer *phishing* de datos. Como no nos queríamos meter a estas páginas (no queríamos que obtuvieran nuestras cuentas bancarias y supieran lo pobres que somos), buscamos una base de datos de páginas web y obtuvimos un *dataset* con algunas características que nos deberían permitir determinar si una página con características similares es o no peligrosa. Tu misión es analizar en detalle este *dataset* y, según sus características, utilizar un modelo predictivo para determinar si las páginas visitadas por Bastián son dañinas o no, y así quitarnos la sospecha o confirmar sus actos deshonestos.

Datasets

Tendrán dos sets de datos. Uno con información de páginas genéricas obtenidas para entrenar un modelo de tipo árbol de decisión, y otro con información sobre las páginas específicas visitadas por Bastián[1]. A continuación, una descripción de los datos por cada entrada:

- **id:** Número identificador de la página.
- **URL of Anchor:** Variable binaria. Indica si los links en la página apuntan a un dominio distinto del tipeado en la barra de direcciones para llegar al sitio.
- **Result:** Clase a predecir. Indica si un sitio es "Legitimate" (1), " Suspicious" (0) o "Phishy" (-1).

[1] Y que no se encuentran en el primer dataset.

- **Query date:** Fecha en que se realizó la evaluación del sitio.
- **having IP Address:** Indica si el dominio contiene una dirección IP.
- **valid IP Adrress:** Variable binaria. Indica si la dirección IP del dominio es válida (1) o no (0).
- **Request URL:** Variable ternaria. Indica si un alto porcentaje de los links son de otros dominios (1), si una cantidad aceptable lo es (0) o si muy pocos lo son (-1)
- **URL Length:** Variable ternaria. Indica si una URL tiene tiene pocos caracteres (1), una cantidad aceptable (0) o muchos caracteres (-1) en la URL.
- **age of domain:** Variable binaria. Indica si el sitio tiene presencia online hace menos de 6 meses (1) o más (-1).
- **SSLfinal State:** Variable ternaria. Describe si la página cumple con los protocolos correspondientes: si usa el protocolo HTTPS y tiene certificados por más de 2 años (1), si usa HTTPS pero los certificados no son de confianza (0) y cualquier otro caso (-1).
- **SFH:** Variable ternaria. Describe cómo la página responde al enviar un formulario de tipo HTTP: si el servidor receptor de los formularios en la página entrega una respuesta vacía (-1), entrega una redirección a otro dominio (0) o cualquier otro caso (1).
- **web traffic:** Variable ternaria. Describe el tráfico recibido por la página en términos de su ranking en la "Alexa database"[2]: mejor que 150.000 (1), peor que 150.000 (0) o no aparece (-1).
- **popUpWindow:** Variable ternaria. Describe como se responde ante un "click derecho". Puede ser que la funcionalidad esté desactivada (-1), que se responda con una alerta (0), o cualquier otro caso (1).

Instrucciones

Paso 1: Análisis de los datos

El *dataset* que obtuvimos con ejemplos de páginas web podría presentar fallos: datos faltantes y datos extremos. Debes mostrar una visualización conveniente de los datos para poder determinar si efectivamente es así, graficando y dejando en evidencia los casos de fallos en los datos.

También debes calcular la media, mediana, moda, varianza y correlación entre la variable Result y todas las demás variables. Deberá imprimir estos valores de forma clara y ordenada.

Paso 2: Limpieza y depuración de datos

Deberás deshacerte de los datos extremos y rellenar los datos faltantes según la media de la columna correspondiente. ¡Ojo! Los datos tienen valores discretos. Si el valor de la media es decimal, deben colocar el valor más cercano.

[2]Base de datos que determina las páginas web más populares en base a su tráfico total.

Paso 3: Clasificar las páginas de Bastián

En base al set de datos que has trabajado anteriormente en esta actividad, deberás entrenar un árbol de decisión para determinar si una página es inofensiva, sospechosa o definitivamente culpable de realizar *phising*. Como requisito, se pide que uses 7 características (ni más, ni menos) para entrenar su modelo. Más allá del número, está permitido experimentar con distintas combinaciones.

Luego, deberás probar tu árbol de decisión ya entrenado sobre el set de páginas que ha visitado Bastián, y entregar el diagnóstico para cada una, guardándolo en la variable Result (con los mismos valores que posee el set de datos de entrenamiento). Finalmente, debes imprimir en consola la cantidad de páginas que ha visitado Bastián que son inofensivas, sospechosas y dañinas.

Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería deben mantener un comportamiento acorde al Código de Honor de la Universidad:

"Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad."

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Ejemplos de actos deshonestos son la copia, el uso de material o equipos no permitidos en las evaluaciones, el plagio, o la falsificación de identidad, entre otros. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica en relación a copia y plagio: Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Si un alumno (grupo) copia un trabajo, se le calificará con nota 1.0 en dicha evaluación y dependiendo de la gravedad de sus acciones podrá tener un 1.0 en todo ese ítem de evaluaciones o un 1.1 en el curso. Además, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir un procedimiento sumario. Por "copia" o "plagio" se entiende incluir en el trabajo presentado como propio, partes desarrolladas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.