



Universidad Nacional
Autónoma de México
Facultad de Ingeniería



Inteligencia Artificial

Proyecto Final - Aplicación Web

Inor Computation



Profesor: Dr. Guillermo Gilberto Molero Castillo

Alumno: Ordiales Caballero, Iñaky

Grupo teoría: 3

Semestre: 2022 - 2

Fecha de entrega: 25 de mayo del 2022.

Índice

1. Introducción	2
1.1. ¿Quiénes somos?	2
1.2. Objetivo	2
2. Requerimientos	3
2.1. Funcionales	3
2.2. No funcionales	3
3. Idea general	3
4. Tecnología y bibliotecas	4
4.1. Python3: Lenguaje principal	4
4.2. Streamlit: Biblioteca para implementar aplicaciones web	4
4.3. HTML5 y CSS: Herramientas para diseño y funcionalidad web	5
5. Borradores	6
6. Implementación y programación	8
6.1. Estructura general y arquitectura	8
6.2. Implementación de acciones y funcionalidades	9
7. Descripción de uso	20
8. Demostración funcionamiento	21
9. Conclusiones	30

1. Introducción



1.1. ¿Quiénes somos?

inor Computation es una aplicación web desarrollada en el lenguaje de programación Python3 y desplegada mediante la utilización de la biblioteca de python Streamlit.

Se realizó como el proyecto final para la asignatura de Inteligencia Artificial de la carrera de Ingeniería en Computación en la Facultad de Ingeniería de la UNAM, por Iñaky Ordiales. En la aplicación se implementaron varios algoritmos de aprendizaje automático para poder ser utilizados con los datos que ingrese el usuario. Además, el usuario podrá elegir los diferentes parámetros a utilizar en cada algoritmo, dando una mayor flexibilidad en la obtención de resultados.



1.2. Objetivo

El objetivo de este proyecto es el de reafirmar los conocimientos adquiridos durante el semestre para la asignatura de Inteligencia Artificial. Esto mediante la implementación de los algoritmos de aprendizaje automático revisados. Adicionalmente como un objetivo secundario se tiene el aprendizaje de frameworks y herramientas extras para su elaboración.

2. Requerimientos

Dentro de los requerimientos del proyecto podemos dividirlos en 2 los requerimientos funcionales que se basan en los servicios que deberá prestar el sistema. Mientras que los requerimientos no funcionales son restricciones del sistema o mejor dicho cómo lo hace.

2.1. Funcionales

- Se deben implementar los algoritmos que se marquen en la bitácora.
- El usuario podrá seleccionar cuál algoritmo utilizar.
- El usuario podrá escoger y cargar su conjunto de datos en formato csv.
- El usuario deberá poder añadir un análisis de los resultados del algoritmo que procese los datos.
- El usuario podrá guardar el resultado del procesamiento en la computadora.

2.2. No funcionales

- La interfaz con el usuario debe ser gráfica e interactiva.
- Se debe de ver limpia y su utilización debe ser sencilla.
- Se utilizará un framework para la elaboración de la interfaz.

3. Idea general

Para la elaboración del proyecto final de la asignatura de Inteligencia Artificial se elaborará una aplicación web que permita a un usuario utilizar diferentes algoritmos para analizar un conjunto de datos. El objetivo del proyecto es demostrar los conocimientos de las diferentes clases de algoritmos de inteligencia artificial revisados a lo largo del curso. Esto de una manera práctica, aplicada y formal que pueda servirnos como un desarrollo que aporte a nuestro currículum profesional.

La forma en la que se aterrizó esto fue mediante una aplicación con múltiples roles de usuarios y en la cual se seleccione un archivo local que contenga los datos en formatos csv y después de aplicarles algún algoritmo y procesamiento (reducción de dimensiones, clasificaciones, etiquetados, etc.) se puedan consultar y almacenar los resultados de manera local. El diseño de la aplicación debería ser sencillo pero atractivo y con colores oscuros para no cansar la vista del usuario. Siendo el enfoque general dos, una alta funcionalidad intuitiva y un diseño agradable.

4. Tecnología y bibliotecas

4.1. Python3: Lenguaje principal



Para toda la lógica de programación de la aplicación, así como el totalidad de su funcionalidad se utilizará el conocido lenguaje de programación Python. Éste además de ser muy flexible y completo, tiene la gran ventaja de todas las bibliotecas existentes que nos permite utilizar. Como hemos visto a lo largo de las primeras prácticas de la asignatura, se tienen muy buenas bibliotecas para el análisis y representación de datos, pero también se cuenta con otras que tienen los algoritmos de inteligencia artificial ya implementados. Es por esto que Python se posiciona como uno de los lenguajes más populares y comunes a la hora de trabajar con inteligencia artificial.

4.2. Streamlit: Biblioteca para implementar aplicaciones web



Streamlit es una biblioteca escrita en Python y para Python la cual fue desarrollada en específico para los ingenieros de Machine Learning. Streamlit convierte los scripts de datos en aplicaciones web de manera muy sencilla e integrada en su totalidad con el lenguaje Python. Este marco de trabajo permite desarrollar una aplicación web con una pequeña curva de aprendizaje y sin necesitar tener gran experiencia en el desarrollo de front-end. Tiene las ventajas de que es código abierto, se refresca la vista de la aplicación automáticamente cada vez que se guarda el script, permite agregar widgets con facilidad y se despliega instantáneamente.

En un planteamiento inicial se vieron otras opciones de framework para utilizar, algunas de las que más se consideraron fueron PyWebIo (descartado por ser muy simple y poco gráfico),

WebPy (descartado por su necesidad de trabajar con otros sistemas como PostgreSQL) y Flask. Éste último fue revisado más a profundidad ya que era más flexible y robusto que Streamlit. Sin embargo, debido a estas características también era más complejo y con una curva de aprendizaje mucho mayor. Al no haber trabajado en el desarrollo de aplicaciones web con anterioridad se decidió quedarnos con la opción que pareciera más sencilla y que se pudiera aprovechar al máximo para no sólo hacer un trabajo superficial. Streamlit ofrece todas las características necesarias para la creación e implementación de una aplicación web. Además, al estar diseñado específicamente para aplicaciones de datos, las herramientas de despliegue de datos están muy bien desarrolladas. Y nos ofrece templates para guiarnos.

4.3. HTML5 y CSS: Herramientas para diseño y funcionalidad web



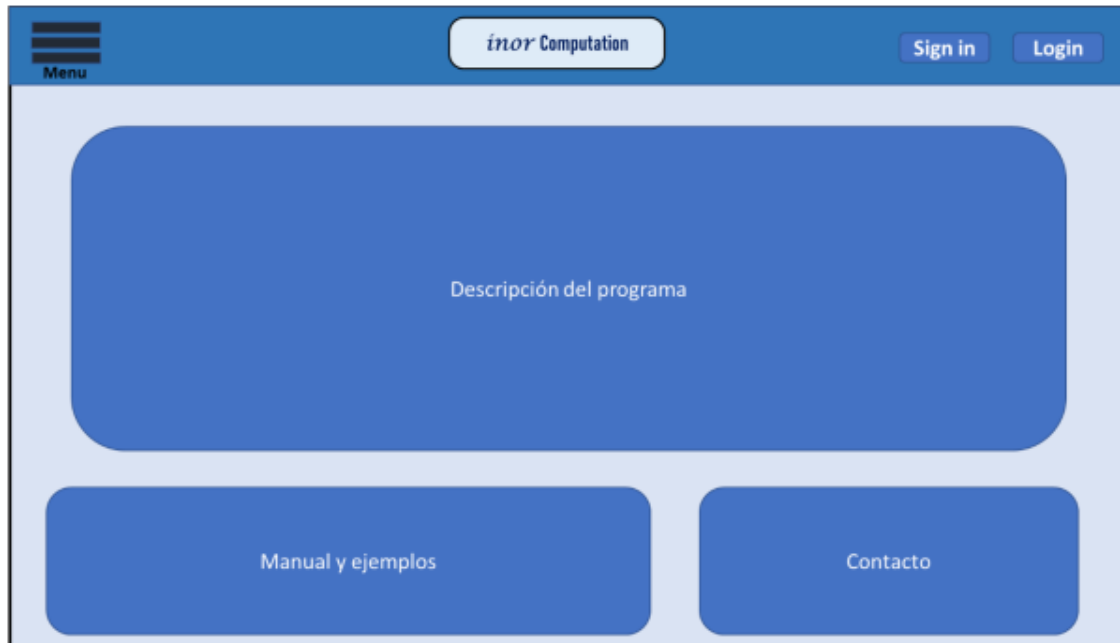
El mundo del internet, de las aplicaciones web y de los navegadores está moldeado mediante el lenguaje de marcado HTML y el estilo CSS. A pesar de que Streamlit nos ayudará a desplegar de manera sencilla nuestra aplicación, es necesario la utilización de estas dos herramientas para definir su forma y estilo. Dentro del mismo framework de Streamlit se tiene que trabajar con HTML para mejores resultados. Y si queremos enserio definir un diseño deberemos utilizar CSS para que se vea bien. Igualmente se ha trabajado poco con ambas herramientas, pero se conoce la sintaxis básica por lo que no deberían de presentar dificultades

**** JavaScript** si bien en un inicio se pensó en usar, finalmente no fue necesario y sólo hubiera complicado más el desarrollo. Python junto con Streamlit fueron capaces de suplir cualquier necesidad.

5. Borradores

Dentro de los borradores originales (que finalmente no se implementaron tal cual, pero sí sirvieron de inspiración) podemos mostrar algunos de los más relevantes.

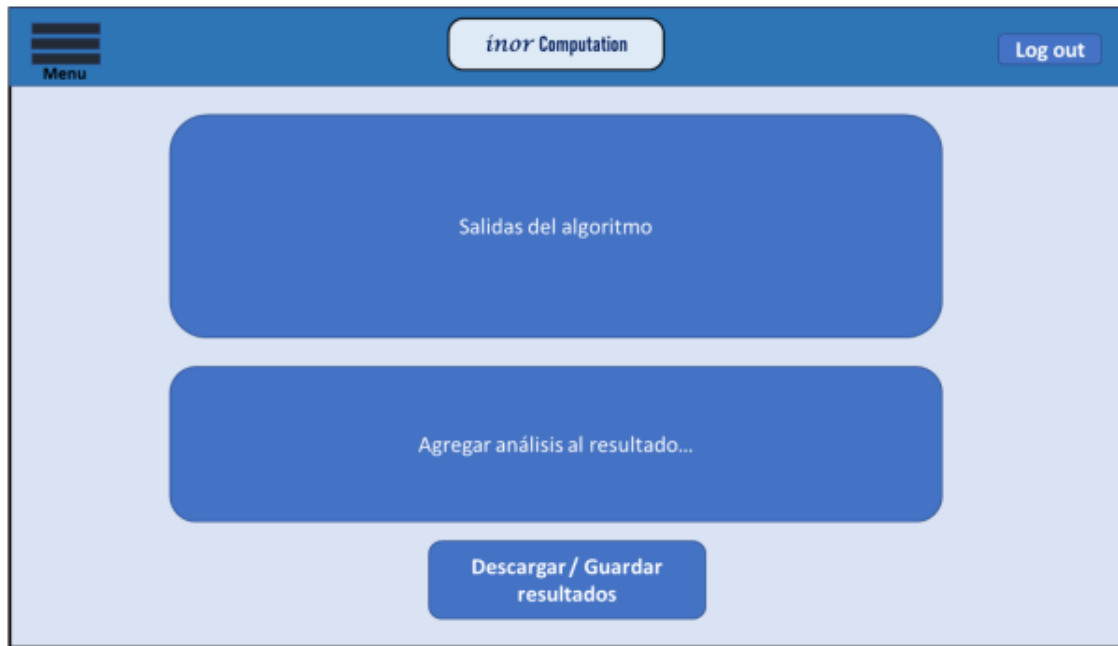
Portada.



Menú.

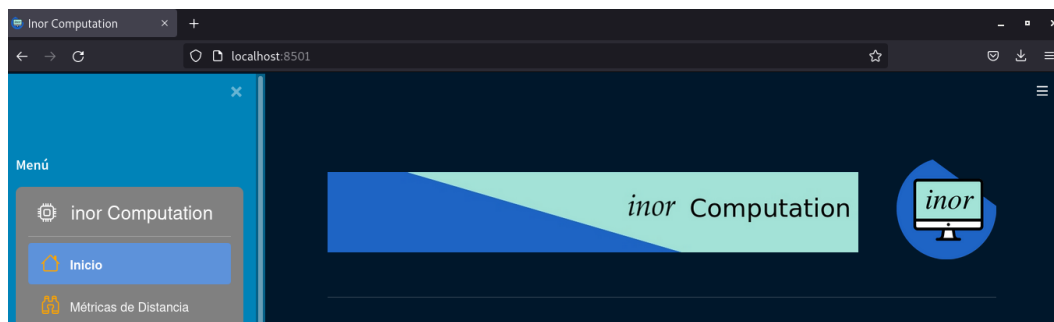


Presentación resultados.



6. Implementación y programación

La programación de la aplicación sí tomó bastante tiempo ya que se decidió hacer un archivo por cada algoritmo, además de los menús, generadores de pdf, páginas de inicio y páginas del creados. Además se enfocó altamente en el diseño visual y personalización de la herramienta y aplicación desarrollada. Esto lo podemos ver a través de la siguiente imagen:



Para todas las páginas de los algoritmos siempre se mostraba el banner y logo de la aplicación hasta arriba, combinando con el fondo azul oscuro y el menú lateral. También se puede apreciar que el icono y nombre de la pestaña son personalizados con el logo y nombre.

6.1. Estructura general y arquitectura

La idea general es que dependiendo del tipo de usuario se podrán realizar diferentes acciones, sobretodo de comentarios y análisis técnicos. Todos podrán analizar datos y generar resultados a través de la interacción mediante botones y widgets con la aplicación web.

Arquitectura de la aplicación web

La arquitectura de una aplicación web describe las interacciones entre aplicaciones, bases de datos y demás sistemas en la web. Se asegura de que el usuario pueda trabajar de manera simultánea en varias aplicaciones de manera correcta. La forma en que el usuario interactuará con la aplicación depende de la arquitectura con la que este elaborada el sistema. La arquitectura deberá incluir todos los subcomponentes de la aplicación, debiendo lidiar con la eficiencia, confianza, escalabilidad, seguridad y robustez.

La arquitectura seleccionada es la de Single Page Architecture (SPA). Las SPA únicamente se componen de HTML, CSS y JavaScript. Su arquitectura se basa en cargar todo lo necesario en una única página, e ir actualizándola dinámicamente con los recursos necesarios y/o añadiéndolos si hicieran falta, a medida que el usuario interactúa con la aplicación o site. Como resultado, la experiencia del usuario es mucho más rápida y fluida. Las SPA's basan su arquitectura en las siguientes características: chunking, controllers, templating, routing, real-time communication, local storage. También se revisó la arquitectura de microservicios como opción si se complicara la SPA.

6.2. Implementación de acciones y funcionalidades

La forma en la que se implementó la funcionalidad fue mediante los algoritmos revisados en clase y la utilización de los widgets proporcionados por streamlit para la elección de parámetros y opciones. Entre estos se usaron botones, cajas de selección, entradas numéricas y áreas de texto.

Además se utilizaron algunas bibliotecas externas como el menú para poder volver más robusta la aplicación. A continuación se muestran todas las opciones que puede tomar el usuario para controlar el flujo del programa y así lograr el funcionamiento esperado.

El menú general:



The screenshot displays the 'inor Computation' application interface. On the left is a sidebar menu titled 'Menú' with the following options: Inicio, Métricas de Distancia, Reglas Asociación - Apriori, Clustering Jerárquico - Ascendente, Clustering Particional - K-means, Regresión Lineal Múltiple - Pronóstico, Regresión Logística - Clasificación, Arbol decisión - Pronóstico, Arbol decisión - Clasificación, Bosques aleatorios - Pronóstico, and **Bosques aleatorios - Clasificación** (highlighted in blue). The main content area on the right has a dark blue background and features the title 'Bosque aleatorio - Clasificación' and 'Árboles de clasificación'. Below this is a table illustrating classification outcomes:

		Predicción	
		Positivos	Negativos
Observación	Positivos	Verdaderos Positivos (VP)	Falsos Positivos (FP)
	Negativos	Falsos Negativos (FN)	Verdaderos Negativos (VN)

Below the table, the section 'Bosques aleatorios' is introduced with the text: 'Una forma de mejorar la generalización de los árboles de decisión combinados. Al combinar lo que en realidad son árboles de decisión para producir un modelo generalizado.'

Después de la elección de la página y algoritmo en el menú la siguiente opción que se presentará es la de subir datos propios o ver un ejemplo. Si se escoge subir datos propios luego el usuario deberá identificar su tipo: técnico o normal.

Elección del usuario:

Implementación algoritmo

Seleccione una opción:

Ejemplo Subir datos

Tipo usuario

Seleccione una opción:

Técnico Normal

Selección de archivo:

Análisis de archivo del usuario

En esta parte el usuario podrá ingresar su archivo para procesar.

Seleccione archivo:

Escoga su archivo formato .csv

Drag and drop file here
Limit 200MB per file • CSV

Browse files

semillas.csv 9.4KB

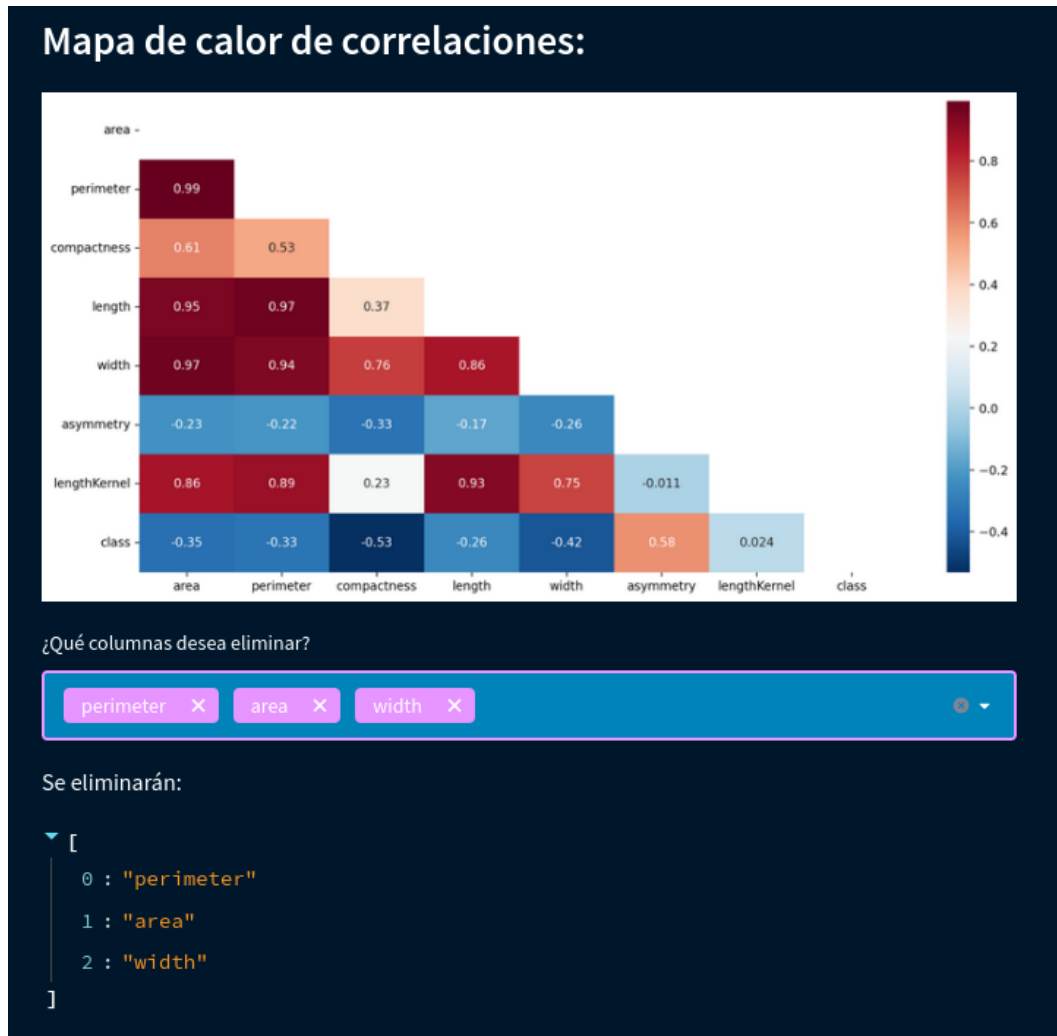
filename: semillas.csv

Los primeros 5 renglones son:

	area	perimeter	compactness	length	width	asymmetry	lengthKernel	class
0	15.2600	14.8400	0.8710	5.7630	3.3120	2.2210	5.2200	1
1	14.8800	14.5700	0.8811	5.5540	3.3330	1.0180	4.9560	1
2	14.2900	14.0900	0.9050	5.2910	3.3370	2.6990	4.8250	1
3	13.8400	13.9400	0.8955	5.3240	3.3790	2.2590	4.8050	1
4	16.1400	14.9900	0.9034	5.6580	3.5620	1.3550	5.1750	1

De igual forma después de subir el archivo es común para todos los algoritmos el mostrarnos el mapa de calor de correlaciones y darnos la opción de escoger que características eliminar.

Selección características:



Ahora sí las siguientes elecciones son propias a cada algoritmo y se muestran a continuación con el nombre al que pertenecen.

Parámetros Métricas:

¿Con qué métrica desea realizar la medición?

☒ Euclidiana
☐ Chebyshev
☐ Manhattan
☐ Minkowski

Métrica elegida: euclidean

Parámetros Apriori:

Soporte mínimo:

0.10 - +

Confianza mínima:

0.15 - +

Levantamiento mínimo:

1.24 - +

Se encontraron un total de 2 reglas:

Reporte de reglas: +

Parámetros Clusters:

Número de cluster:

3

Datos con su etiqueta del cluster jerárquico:

	compactness	length	asymmetry	lengthKernel	class	clusterH
0	0.8710	5.7630	2.2210	5.2200	1	1
1	0.8811	5.5540	1.0180	4.9560	1	2
2	0.9050	5.2910	2.6990	4.8250	1	2
3	0.8955	5.3240	2.2590	4.8050	1	2
4	0.9034	5.6580	1.3550	5.1750	1	2
5	0.8951	5.3860	2.4620	4.9560	1	2
6	0.8799	5.5630	3.5860	5.2190	1	2
7	0.8911	5.4200	2.7000	5.0000	1	2
8	0.8747	6.0530	2.0400	5.8770	1	1
9						

Centroides de cada grupo:

	compactness	length	asymmetry	lengthKernel	class
0	0.8425	5.2740	5.4246	5.1533	2.8571
1	0.8802	6.0565	3.3568	5.8575	1.7473
2	0.8757	5.3161	3.1654	5.0158	1.8312

Parámetros Clasificación:

¿Cuál es la variable de clase?

Species

Validación del modelo

	0	1	2
0	32	0	0
1	0	18	0
2	0	0	15

Modelo:

ECUACION:

Coefficientes: $[[-0.860438, 0.975326, -0.010742, 0.001894, 0.173814, -0.613248], [-0.428905, -0.993256, 0.11425, 0.0052, -0.257918, 0.241954], [1.289344, 0.01793, -0.103509, -0.007094, 0.084105, 0.371294]]$

Intercepto: $[0.020416, -0.015376, -0.00504]$

Parámetros Pronóstico:

¿Cuál es la variable pronóstico?

hora

Obtención de los coeficientes, intercepto, error y Score

DATOS DEL MODELO:

Coeficientes: $[[-0.168586, 0.048439, 0.053547, 1.055769, 0.007378, -0.067115, 0.03702, 0.013265]]$

Intercepto: $[11.388258]$

Residuo: 16.055288

MSE: 35.78989

RMSE: 5.982465

Score (Bondad de ajuste): 0.218899

Parámetros Bosque:

¿Cuál es la variable pronóstico?

count

Número de árboles estimadores:

100

Profundidad:

8


Mínimo para decisión:

4

Mínimo para hoja:

2

Selección Árbol:

 Selección del árbol n para mostrar

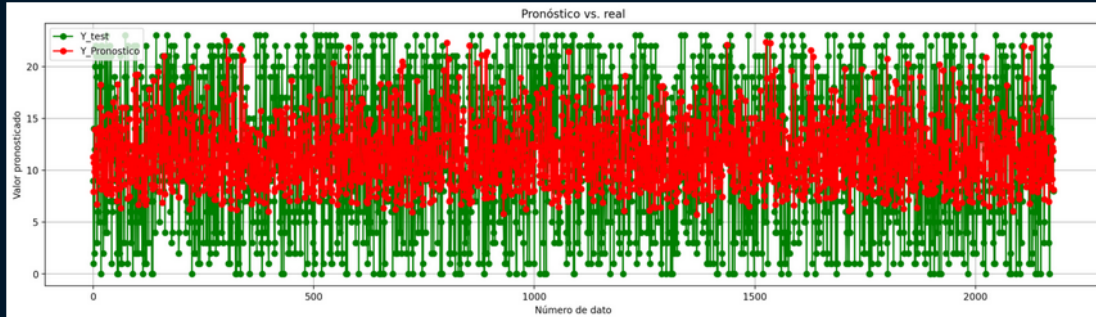
Número del árbol que mostrar:

1

Ejemplos de resultados.

Gráfico Pronósticos:

Visualización gráfica



Reglas Apriori:

Se encontraron un total de 2 reglas:

Reporte de reglas:

[1]

Regla: frozenset({'dishwashing liquid/detergent', 'individual meals'})

Soporte: 0.1021671826625387

Confianza: 0.37078651685393255

Lift: 1.2740855845087258

[2]

Regla: frozenset({'toilet paper', 'waffles'})

Soporte: 0.11042311661506708

Confianza: 0.39051094890510946

Lift: 1.278395640165713

Finalmente para el usuario técnico este puede realizar un comentario o análisis de los resultados en un área de texto. Estos comentarios se añadirán al reporte en formato PDF que se generará y al apretar el botón de Listo se mostrará un enlace para poder descargar los documentos correspondientes.

Interpretación antes:

Escriba las interpretaciones:

El grupo 1 tiene la mayor compactitud, largo y largo del nucleo.

Listo

Interpretación después:

Escriba las interpretaciones:


El grupo 1 tiene la mayor compactitud, largo y largo del nucleo.

Listo

[Download file](#)

Ejemplos archivos resultantes:

inor Computation



Reglas Apriori

```
[1]
Regla: frozenset([' dishwashing liquid/detergent', ' individual meals'])
Soporte: 0.1021671826625387
Confianza: 0.37078651685393255
Lift: 1.2740855845087258
-----

[2]
Regla: frozenset([' toilet paper', ' waffles'])
Soporte: 0.11042311661506708
Confianza: 0.39051094890510946
Lift: 1.278395640165713
-----
```

Comentarios técnicos:

Las reglas parecen correctas.

inor Computation

Clustering Jerárquico

	compactness	length	asymmetry	lengthKernel
clusterH				
0	0.864922	5.357237	3.741886	5.092791
1	0.882894	6.159662	3.618592	6.025310

Comentarios técnicos:

El grupo 1 tiene la mayor compactitud, largo y largo del núcleo.

inor Computation



Regresión Lineal Múltiple

DATOS DEL MODELO:

Coefficientes: $[-0.168586, 0.048439, 0.053547, 1.055769, 0.007378, -0.067115, 0.03702, 0.013265]$

Intercepto: $[11.388258]$

Residuo: 16.055288

MSE: 35.78989

RMSE: 5.982465

Score (Bondad de ajuste): 0.218899

Comentarios técnicos:

El modelo no sirve, tiene un ajuste muy bajo y no se debe utilizar.

7. Descripción de uso



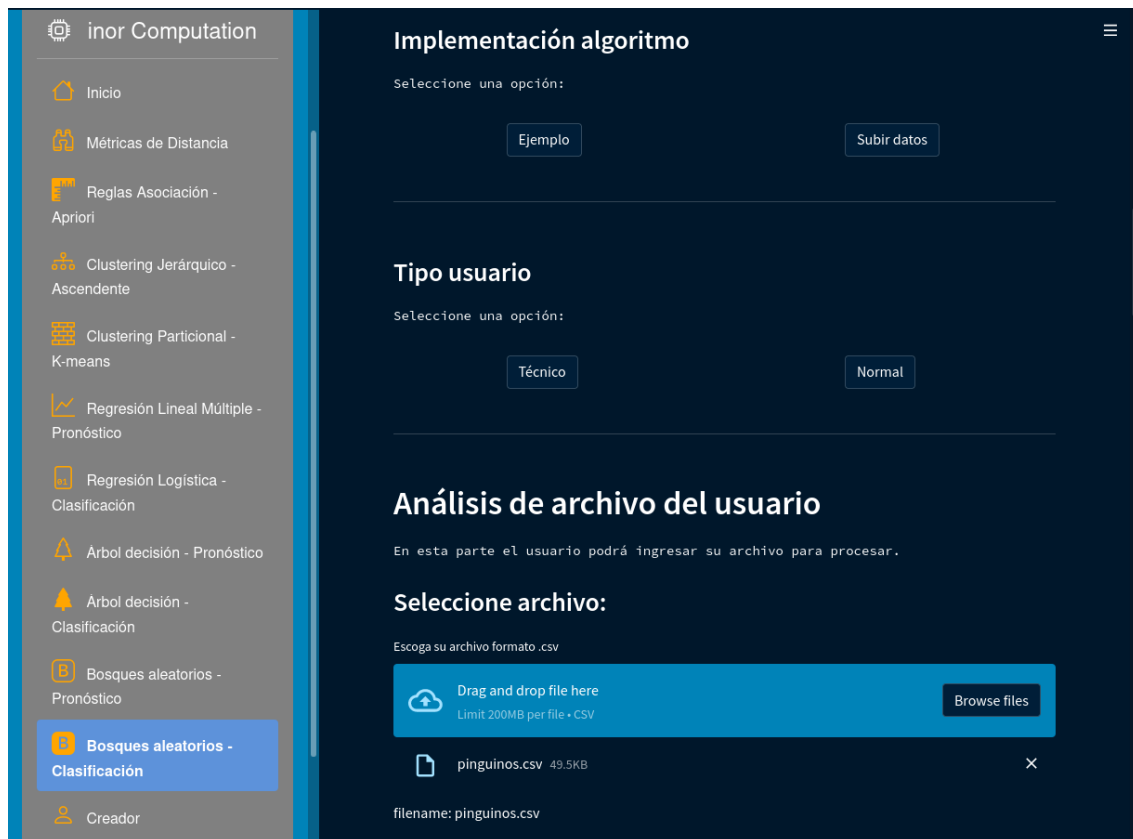
Por default al ingresar a la aplicación a través del navegador se debe desplegar el menú lateral donde aparecerán las opciones de la aplicación: inicio, métricas, ..., creador. Seleccionando cada una de ellas se le mostrará la información correspondiente a dicha sección. En el caso de que el menú no se encuentre abierto, en la parte superior izquierda dentro de la ventana del navegador aparece una flecha de la forma '>' con la que se podrá abrir el menú. En caso de quererlo achicar, el menú tiene una cruz del lado superior derecho que lo cerrará tal y como se muestra en la imagen superior.

Dependiendo de la selección de contenido realizada en el menú, la aplicación actualizará la pantalla para mostrar la información del tema o algoritmo escogido. En el caso de ser un algoritmo (no la página de inicio o la del creador) al recorrer hacia abajo la página, aparecerán dos botones, uno para escoger ver un ejemplo y el otro para ingresar nuestros propios datos al sistema para analizarlos con la herramienta. Si escoge la demostración esta se llevará a cabo totalmente en automático con datos ya establecidos, es una forma de que el usuario vea de qué forma analiza los datos el algoritmo específico y qué resultados nos devuelve. Por el segundo camino de ingresar nuestros propios datos, se nos preguntará el tipo de usuario 'técnico o normal' y dependiendo de la elección se podrá analizar los resultados y descargarlos o sólo verlos en pantalla. Para ambos tipos de usuario se le darán diversos parámetros y elecciones durante la aplicación del algoritmo y análisis de datos. Se irán mostrando ordenadamente los datos en pantalla y se podrá navegar y modificar a través de botones, y demás widgets proporcionados por streamlit.

8. Demostración funcionamiento

Para demostrar el funcionamiento de la aplicación web, se ha decidido hacerlo mediante las capturas de un ejemplo completo de ejecución. En este caso se mostrará la clasificación mediante un bosque aleatorio. Los datos utilizados son registros de pingüinos y sus características físicas que se obtuvieron de un centro de estudios científicos sobre la vida salvaje y la naturales en la Antártica. El objetivo sería clasificarlos dentro de una de las tres especies según las demás características seleccionadas. A continuación se muestran las diferentes partes del funcionamiento:

Después de la descripción del algoritmo (que omitimos mostrar en este ejemplo) se le da al usuario sus primeras elecciones para el funcionamiento. Primero puede escoger entre ver un ejemplo el cual es un demo donde se muestra el procesamiento completo de unos datos ya escogidos, pero no se puede seleccionar ningún ajuste. La otra opción (la que se escogió) es la de subir nuestros propios datos. Posteriormente el usuario elegirá su tipo: técnico si quiere realizar una descripción/comentarios/análisis de los resultados y generar un pdf con ellos, o normal si sólo quiere procesar y ver el resultado como tal. Nosotros escogimos técnico para el ejemplo. Lo siguiente es subir el archivo correspondiente con los datos: **pinguinos.csv**. El selector de archivos sólo permite ingresar un único archivo con extensión csv a la vez.



Lo primero que se muestra una vez cargado el archivo son los 5 primeros renglones de éste para que podamos corroborar que sí son los datos deseados. Después también nos muestra la

información de cada una de las columnas que se tienen.

Los primeros 5 renglones son:

	Unnamed: 0	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Cl
0	1	PAL0708		Adelie Penguin (Pygosceli...	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Y
1	2	PAL0708		Adelie Penguin (Pygosceli...	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Y
2	4	PAL0708		Adelie Penguin (Pygosceli...	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Y
3	5	PAL0708		Adelie Penguin (Pygosceli...	Anvers	Torgersen	Adult, 1 Egg Stage	N3A2	Y
4	6	PAL0708		Adelie Penguin (Pygosceli...	Anvers	Torgersen	Adult, 1 Egg Stage	N4A1	Y

Información datos:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 325 entries, 0 to 324
Data columns (total 17 columns):
#   Column                                Non-Null Count  Dtype
---  ---                                -
0   Unnamed: 0                            325 non-null    int64
1   studyName                            325 non-null    object
2   Sample Number                        325 non-null    int64
3   Species                              325 non-null    object
4   Region                              325 non-null    object
5   Island                              325 non-null    object
6   Stage                               325 non-null    object
7   Individual ID                        325 non-null    object
8   Clutch Completion                    325 non-null    object
9   Date Egg                            325 non-null    object
10  Culmen Length (mm)                   325 non-null    float64
11  Culmen Depth (mm)                   325 non-null    float64
12  Flipper Length (mm)                  325 non-null    float64
13  Body Mass (g)                       325 non-null    float64
14  Sex                                  325 non-null    object
15  Delta 15 N (o/oo)                   325 non-null    float64
16  Delta 13 C (o/oo)                   325 non-null    float64
dtypes: float64(6), int64(2), object(9)
memory usage: 43.3+ KB
```

Más adelante se empiezan a analizar los datos mostrando en pantalla la descripción de ellos (cuenta, media, desviación, cuartiles, etc.) y la matriz de correlación para la selección de características.

Descripción de los datos:

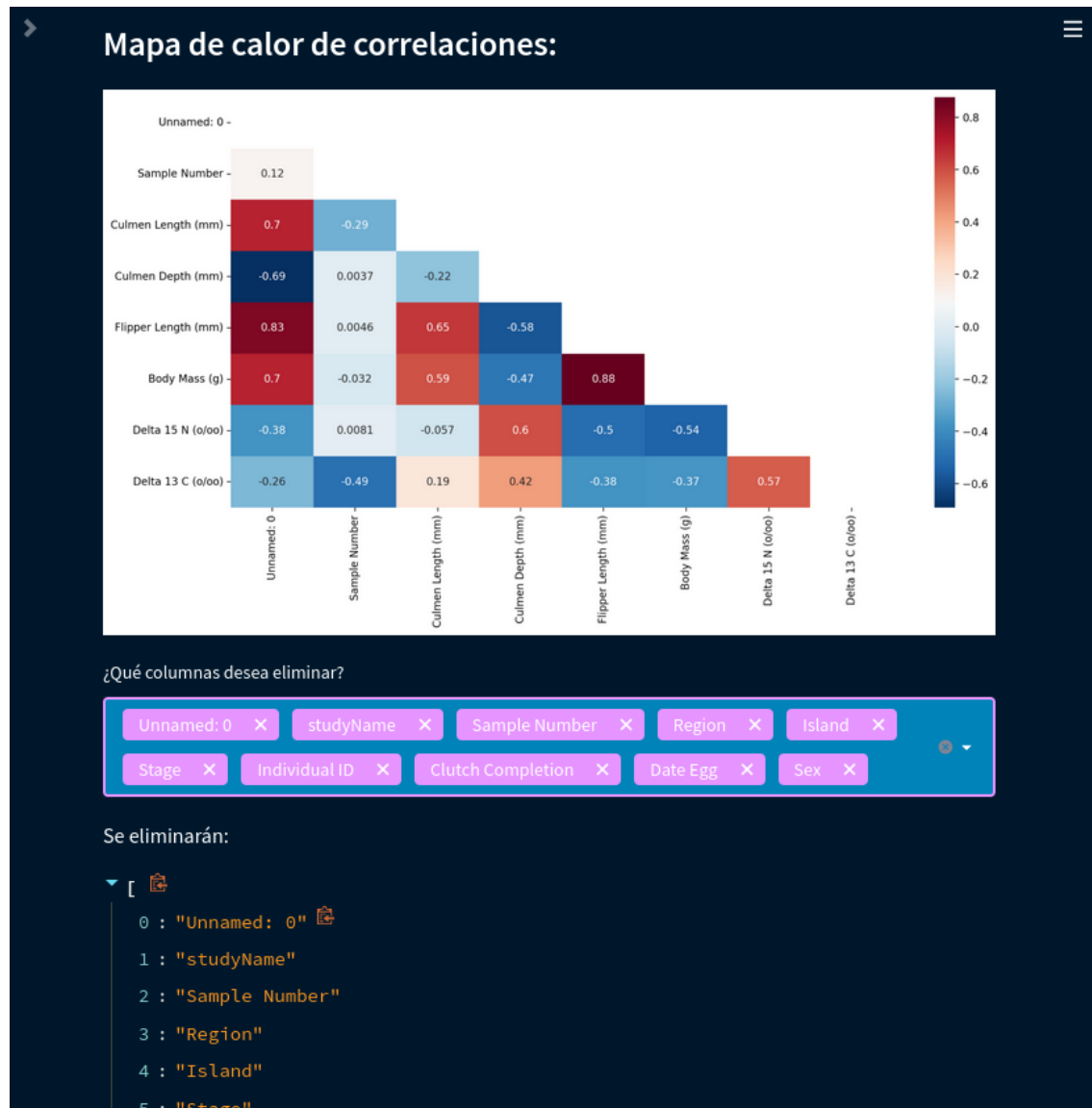
	Unnamed: 0	Sample Number	Culmen Length (mm)	Culmen Depth (mm)	Flipper Length (mm)	Body Mass (g)	Delta 15 N (o/oo)
count	325.0000	325.0000	325.0000	325.0000	325.0000	325.0000	325.0000
mean	175.6462	64.7569	44.0548	17.1231	201.2646	4,216.0000	
std	96.3089	40.2693	5.4722	1.9659	13.9646	808.8592	
min	1.0000	1.0000	32.1000	13.1000	172.0000	2,700.0000	
25%	94.0000	31.0000	39.5000	15.6000	190.0000	3,550.0000	
50%	175.0000	60.0000	44.9000	17.3000	197.0000	4,050.0000	
75%	259.0000	97.0000	48.7000	18.6000	213.0000	4,800.0000	
max	343.0000	157.0000	59.6000	21.5000	221.0000	6,300.0000	

Matriz de correlación:

	Unnamed: 0	Sample Number	Culmen Length (mm)	Culmen Depth (mm)	Flipper Length (mm)	Body Mass (g)	Delta 15 N (o/oo)	Delta 13 C (o/oo)
Unnamed: 0	1.0000	0.1161	0.6977	-0.6918	0.8282	0.6963		
Sample Number	0.1161	1.0000	-0.2861	0.0037	0.0046	-0.0320		
Culmen Length (mm)	0.6977	-0.2861	1.0000	-0.2229	0.6479	0.5913		
Culmen Depth (mm)	-0.6918	0.0037	-0.2229	1.0000	-0.5759	-0.4711		
Flipper Length (mm)	0.8282	0.0046	0.6479	-0.5759	1.0000	0.8773		
Body Mass (g)	0.6963	-0.0320	0.5913	-0.4711	0.8773	1.0000		
Delta 15 N (o/oo)	-0.3757	0.0081	-0.0571	0.6003	-0.5038	-0.5411	1.0000	
Delta 13 C (o/oo)	-0.2616	-0.4011	0.1982	0.4208	-0.2768	-0.2720	-0.5411	1.0000

Como es complicado ver numéricamente toda la matriz de correlaciones se añade un mapa de calor que muestra la fuerza de la correlación. Inmediatamente después se nos da la opción de seleccionar la columna o características que se desean eliminar. Podemos añadir tantas como queramos, dejando sólo nuestras variables independiente y la variable pronóstico.

Abajo se nos muestra en forma de lista las columnas que eliminaremos, para poder checar que no se nos pase ninguna.



Una vez seleccionando las características se mostrará en pantalla la nueva matriz con la que se trabajará, la cual ya tiene sólo las características seleccionadas junto con la variable para clasificar.

Inmediatamente después se nos da la opción de seleccionar la variable de clase que deseemos clasificar con el bosque aleatorio e igualmente se nos pide ingresemos diferentes parámetros. Los parámetros que podemos ajustar son:

- Número de estimadores: cuántos árboles para el bosque.
- Profundidad máxima: qué tan profundo (número de subárboles) puede tener.
- Mínimo decisión: cuántos elementos mínimos en un nodo de decisión.
- Mínimo hoja: cuántos elementos mínimos en un nodo hoja (o se poda).

Matriz de datos reducida dimensionalidad:

	Species	Culmen Length (mm)	Culmen Depth (mm)	Flipper Length (mm)
0	Adelie Penguin (Pygoscelis...	39.5000	17.4000	
1	Adelie Penguin (Pygoscelis...	40.3000	18.0000	
2	Adelie Penguin (Pygoscelis...	36.7000	19.3000	
3	Adelie Penguin (Pygoscelis...	39.3000	20.6000	
4	Adelie Penguin (Pygoscelis...	38.9000	17.8000	
5	Adelie Penguin (Pygoscelis...	39.2000	19.6000	
6	Adelie Penguin (Pygoscelis...	34.6000	21.1000	
7	Adelie Penguin (Pygoscelis...	38.7000	19.0000	
8	Adelie Penguin (Pygoscelis...	42.5000	20.7000	
9				

¿Cuál es la variable de clase?

Species

Número de árboles estimadores:

101

Profundidad:

8

Mínimo para decisión:

4

Mínimo para hoja:

2

Con todo esto definido se muestra en pantalla los resultados numéricos del modelo (matriz de confusión, métricas, exactitud, importancia, etc.)

Modelo:

DATOS DEL MODELO:

Criterio: gini

MATRIZ CLASIFICACION:

Clasificación	Adelie Penguin (Pygoscelis adeliae)	Chinstrap penguin (Pygoscelis antarctica)	Gentoo penguin (Pygoscelis papua)
Real			
Adelie Penguin (Pygoscelis adeliae)	29	0	0
Chinstrap penguin (Pygoscelis antarctica)	1	12	0
Gentoo penguin (Pygoscelis papua)	0	0	23

MÉTRICAS:

	precision	recall	f1-score	support
Adelie Penguin (Pygoscelis adeliae)	0.97	1.00	0.98	29
Chinstrap penguin (Pygoscelis antarctica)	1.00	0.92	0.96	13
Gentoo penguin (Pygoscelis papua)	1.00	1.00	1.00	23
accuracy			0.98	65
macro avg	0.99	0.97	0.98	65
weighted avg	0.99	0.98	0.98	65

EXACTITUD:

0.9846153846153847

Importancia de las variables:

	Variable	Importancia
0	Culmen Length (mm)	0.2626
2	Flipper Length (mm)	0.2258
1	Culmen Depth (mm)	0.1711
5	Delta 13 C (o/oo)	0.1566
3	Body Mass (g)	0.1089
4	Delta 15 N (o/oo)	0.0751

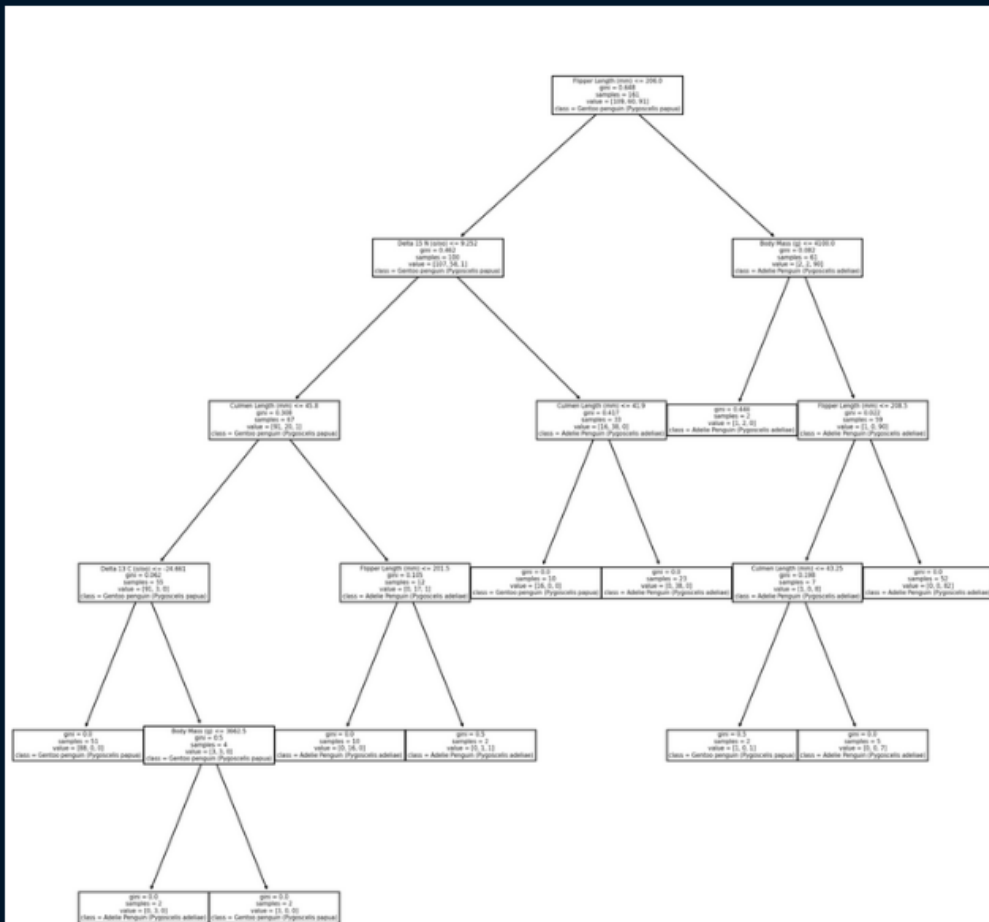
Como lo hicimos en clase, también el usuario tendrá la opción de escoger un árbol del ensamble para ver cómo se construyó. En este caso escogemos el árbol número 33 y se nos muestra su imagen.

Selección del árbol n para mostrar

Número del árbol que mostrar:

33

Árbol creado:



Ya casi para terminar se nos mostrarán los datos del árbol, las reglas de decisión que utiliza. Estas como pueden llegar a ser mucho se puede colapsar el texto para evitar ocupe mucho espacio, pero ahora no fue necesario y se muestran completas. Justo debajo de las reglas tenemos el área de texto para añadir nuestros comentarios. Una vez los agregamos le damos al botón listo y aparece el link de descarga.

```

Reporte de reglas:

|--- Flipper Length (mm) <= 206.00
|   |--- Delta 15 N (o/oo) <= 9.25
|   |   |--- Culmen Length (mm) <= 45.80
|   |   |   |--- Delta 13 C (o/oo) <= -24.66
|   |   |   |   |--- class: 0.0
|   |   |   |   |--- Delta 13 C (o/oo) > -24.66
|   |   |   |   |   |--- Body Mass (g) <= 3662.50
|   |   |   |   |   |   |--- class: 1.0
|   |   |   |   |   |   |--- Body Mass (g) > 3662.50
|   |   |   |   |   |   |   |--- class: 0.0
|   |   |   |--- Culmen Length (mm) > 45.80
|   |   |   |   |--- Flipper Length (mm) <= 201.50
|   |   |   |   |   |--- class: 1.0
|   |   |   |   |   |--- Flipper Length (mm) > 201.50
|   |   |   |   |   |   |--- class: 1.0
|   |   |--- Delta 15 N (o/oo) > 9.25
|   |   |   |--- Culmen Length (mm) <= 41.90
|   |   |   |   |--- class: 0.0
|   |   |   |   |--- Culmen Length (mm) > 41.90
|   |   |   |   |   |--- class: 1.0
|--- Flipper Length (mm) > 206.00
|   |--- Body Mass (g) <= 4100.00
|   |   |--- class: 1.0
|   |   |--- Body Mass (g) > 4100.00
|   |   |   |--- Flipper Length (mm) <= 208.50
|   |   |   |   |--- Culmen Length (mm) <= 43.25
|   |   |   |   |   |--- class: 0.0
|   |   |   |   |   |--- Culmen Length (mm) > 43.25
|   |   |   |   |   |   |--- class: 2.0
|   |   |   |--- Flipper Length (mm) > 208.50
|   |   |   |   |--- class: 2.0
  
```

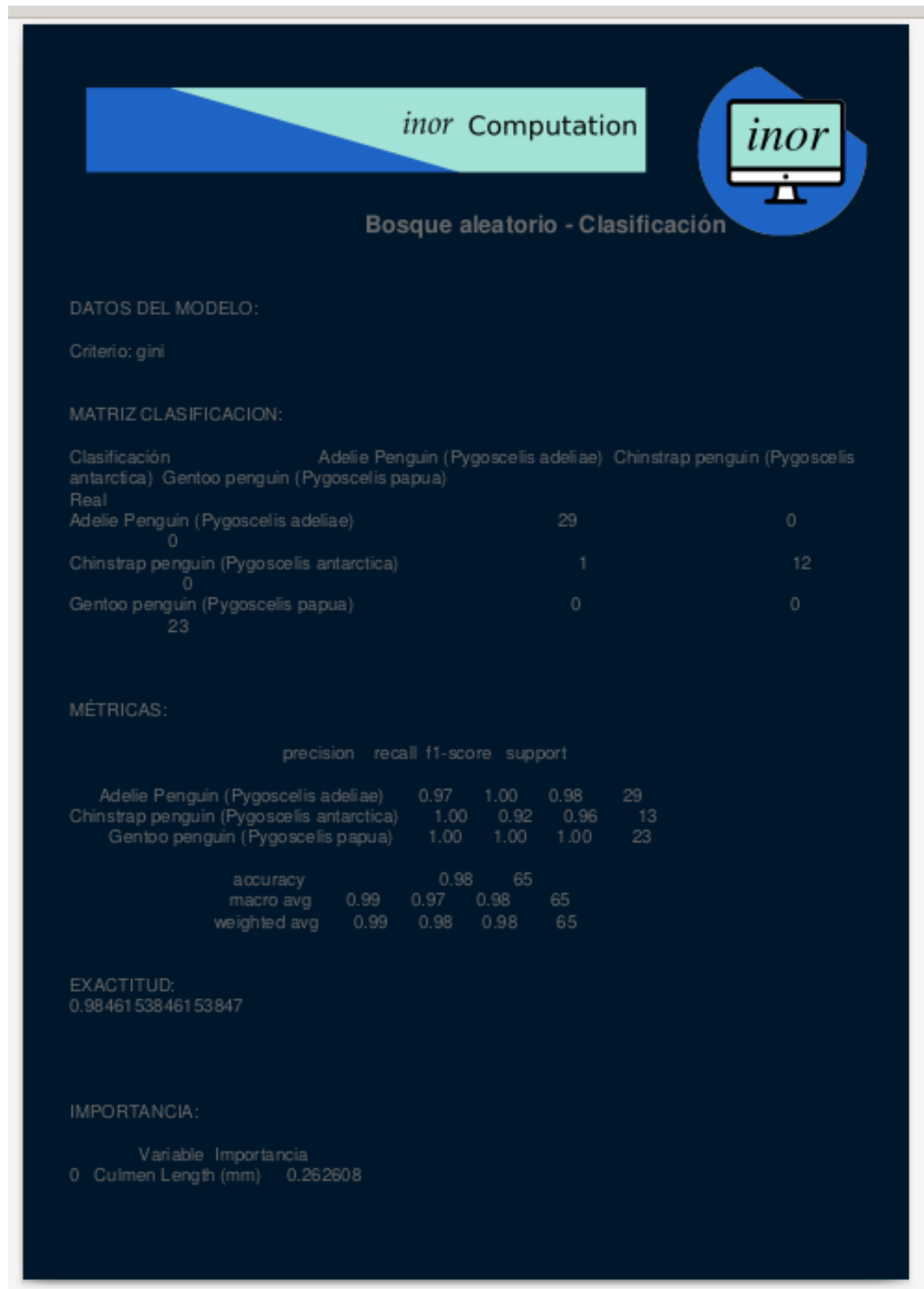
Escriba las interpretaciones:

Con un 98% de exactitud el modelo parece muy bueno, habría que probar por el sobre ajuste pero no creemos se esté dando. Simplemente es sencillo diferenciar a los pingüinos por sus características físicas.

Listo

[Download file](#)

Descargando el resultado se obtiene un PDF con las siguientes dos páginas (la primera personalizada por inorComputation).



```

2 Flipper Length (mm) 0.225761
1 Culmen Depth (mm) 0.171071
5 Delta 13 C (o/oo) 0.156586
3 Body Mass (g) 0.108863
4 Delta 15 N (o/oo) 0.075112

```

REPORTE:

```

|-- Flipper Length (mm) <= 206.00
| |-- Delta 15 N (o/oo) <= 9.25
| | |-- Culmen Length (mm) <= 45.80
| | | |-- Delta 13 C (o/oo) <= -24.66
| | | | |-- class: 0.0
| | | |-- Delta 13 C (o/oo) > -24.66
| | | | |-- Body Mass (g) <= 3662.50
| | | | | |-- class: 1.0
| | | | |-- Body Mass (g) > 3662.50
| | | | | |-- class: 0.0
| | | |-- Culmen Length (mm) > 45.80
| | | |-- Flipper Length (mm) <= 201.50
| | | | |-- class: 1.0
| | | |-- Flipper Length (mm) > 201.50
| | | | |-- class: 1.0
| |-- Delta 15 N (o/oo) > 9.25
| | |-- Culmen Length (mm) <= 41.90
| | | |-- class: 0.0
| | |-- Culmen Length (mm) > 41.90
| | | |-- class: 1.0
|-- Flipper Length (mm) > 206.00
| |-- Body Mass (g) <= 4100.00
| | |-- class: 1.0
| |-- Body Mass (g) > 4100.00
| | |-- Flipper Length (mm) <= 208.50
| | | |-- Culmen Length (mm) <= 43.25
| | | | |-- class: 0.0
| | | |-- Culmen Length (mm) > 43.25
| | | | |-- class: 2.0
| | |-- Flipper Length (mm) > 208.50
| | | |-- class: 2.0

```

Comentarios técnicos:

Con un 98% de exactitud el modelo parece muy bueno, habría que probar por el sobre ajuste pero no creemos se esté dando. Simplemente es sencillo diferenciar a los pingüinos por sus características físicas.

Con esto concluiría la demostración del funcionamiento de la aplicación web.

9. Conclusiones

La realización de este proyecto realmente fue un desafío personalmente, a mi parecer es uno de los trabajos con mayor dificultad que he realizado a lo largo de la carrera. Esto debido a las restricciones y limitaciones del tiempo debido a todo el demás trabajo de la asignatura de Inteligencia Artificial, así como también de las demás materias del plan de estudios. Sin embargo, considero que la parte más complicada fue el desarrollar nuestra interfaz gráfica con la que el usuario pudiera interactuar de forma amigable e intuitiva. Anteriormente nunca había desarrollado una aplicación como esta, por lo que la curva de aprendizaje sí fue complicada de lograr.

Ya teniendo terminado el proyecto, creo que fue una gran realización que cubre adecuadamente con los objetivos y requerimientos planteados para este proyecto. El programa implementó todos los algoritmos requeridos, se le dio la capacidad de seleccionar características, alterar parámetros al usuario y realizar su análisis técnico, y se agregaron cosas extras como la generación de un PDF con los resultados más relevantes. Se desarrolló un gran esfuerzo para lograr todo esto, pero finalmente con el resultado final puedo establecer que valió la pena. La utilización de la biblioteca Streamlit como el framework de front-end facilitó mucho la implementación gráfica, sin embargo aún se debió aprender a aprovechar al máximo la biblioteca. Lo último que destacar del proyecto realizado, inor Computation, es lo altamente personalizado que está el diseño. La página es muy bonita y amigable a la vista, así como totalmente personalizada.

inor Computation

