



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería  
Informática**

**Herramienta docente para la  
visualización en Web de  
algoritmos de aprendizaje  
Semi-Supervisado  
Documentación técnica**



Presentado por David Martínez Acha  
en Universidad de Burgos — 5 de febrero  
de 2023

Tutor: Álgvar Arnaiz González  
Cotutor: César Ignacio García Osorio



---

# Índice general

---

|   |            |
|---|------------|
| <b>Índice general</b>                                   | <b>i</b>   |
| <b>Índice de figuras</b>                                | <b>iii</b> |
| <b>Índice de tablas</b>                                 | <b>iv</b>  |
| <b>Apéndice A Plan de Proyecto Software</b>             | <b>1</b>   |
| A.1. Introducción . . . . .                             | 1          |
| A.2. Planificación temporal . . . . .                   | 1          |
| A.3. Estudio de viabilidad . . . . .                    | 5          |
| <b>Apéndice B Especificación de Requisitos</b>          | <b>7</b>   |
| B.1. Introducción . . . . .                             | 7          |
| B.2. Objetivos generales . . . . .                      | 7          |
| B.3. Catalogo de requisitos . . . . .                   | 7          |
| B.4. Especificación de requisitos . . . . .             | 7          |
| <b>Apéndice C Especificación de diseño</b>              | <b>9</b>   |
| C.1. Introducción . . . . .                             | 9          |
| C.2. Diseño de datos . . . . .                          | 9          |
| C.3. Diseño procedimental . . . . .                     | 9          |
| C.4. Diseño arquitectónico . . . . .                    | 9          |
| C.5. Diseño de la Web . . . . .                         | 9          |
| <b>Apéndice D Documentación técnica de programación</b> | <b>13</b>  |
| D.1. Introducción . . . . .                             | 13         |
| D.2. Estructura de directorios . . . . .                | 13         |

|  |           |
|--|-----------|
| D.3. Manual del programador . . . . .                            | 13        |
| D.4. Compilación, instalación y ejecución del proyecto . . . . . | 13        |
| D.5. Pruebas del sistema . . . . .                               | 13        |
| <b>Apéndice E Documentación de usuario</b>                       | <b>15</b> |
| E.1. Introducción . . . . .                                      | 15        |
| E.2. Requisitos de usuarios . . . . .                            | 15        |
| E.3. Instalación . . . . .                                       | 15        |
| E.4. Manual del usuario . . . . .                                | 15        |
| <b>Bibliografía</b>  | <b>17</b> |

---

## Índice de figuras

---

|   |    |
|---|----|
| C.1. Página inicial de la Web. . . . .              | 10 |
| C.2. Página de configuración del algoritmo. . . . . | 11 |
| C.3. Página de ejecución del algoritmo. . . . .     | 12 |

---

# Índice de tablas

---

|   |   |
|---|---|
| B.1. CU-1 Nombre del caso de uso. . . . . | 8 |
|---|---|

## *Apéndice A*

---

# **Plan de Proyecto Software**

---

### **A.1. Introducción**

En el presente apartado de los anexos se analizará la gestión del proyecto software desarrollado. Este proyecto será organizado mediante la metodología Scrum en la que el trabajo estará dividido en Sprints. Por cada Sprint se realiza una reunión para la revisión del avance y los objetivos para el siguiente. Con esta metodología se mantendrá en todo momento lo que se conoce como **Product Backlog** que es una lista de las tareas a realizar. Esta lista será actualizada, en principio, en cada reunión para así mantener un desarrollo constante. Las reuniones en un principio se realizan cada dos semanas, intensificando a cada semana en el momento del inicio del periodo temporal del segundo cuatrimestre.

El objetivo de este plan es servir como herramienta para registrar el avance del proyecto y también para poder cumplir con el objetivo final del desarrollo.

### **A.2. Planificación temporal**

La planificación temporal se llevará a cabo mediante Sprint de 2 semanas. En la presente sección se comentará el desarrollo realiza en cada uno de ellos.

## Sprint 0

Desde el punto de vista temporal corresponde desde el inicio del curso del primer cuatrimestre académico (septiembre) hasta el Sprint 1. El día 15 de septiembre se tuvo la primera reunión con los tutores sobre el trabajo presente donde se establecieron las líneas generales y temática sobre el mismo.

Se creó el repositorio del TFG en Github: <https://github.com/dma1004/TFG-SemiSupervisado> y se añadió la plantilla de la documentación.

## Sprint 1

Corresponde con el periodo temporal del 5 al 19 de octubre de 2022.

El mismo día 5 tuvo lugar una reunión de seguimiento del trabajo. Durante el sprint se realizaron unos arreglos de la plantilla y una lectura de conceptos teóricos para posteriormente añadirlos a la documentación. Concretamente se crearon las tareas «Añadir conceptos teóricos aprendizaje» y «Trabajos relacionados» a día 9 de octubre.

## Sprint 2

Corresponde con el periodo temporal del 19 de octubre al 2 de noviembre de 2022.

Durante el sprint se implementó un prototipo del algoritmo Self-Training en el que posteriormente se hicieron unas correcciones en el código. También se comenzó con la redacción de conceptos teóricos (tarea «In progress»), concretamente, sobre el aprendizaje automático.

## Sprint 3

Corresponde con el periodo temporal del 16 al 30 de noviembre de 2022.

Durante el sprint se aumentaron los conceptos teóricos sobre el aprendizaje supervisado, no supervisado y semi-supervisado. Se refactorizó el prototipo para su documentación (PEP), evitar datos duplicados y modularizando el código.

La memoria fue parcialmente modificada basándose en las correcciones propuestas de los tutores.



## Sprint 4

Corresponde con el periodo temporal del 25 de enero al 1 de febrero de 2023. En este momento las duraciones de los Sprints cambiaron a una semana, iniciando así el periodo temporal real del desarrollo del proyecto (segundo cuatrimestre)

Durante el sprint se retomaron las tareas y el desarrollo general del proyecto. Se mejoró el algoritmo de **SelfTraining** que estaba como prototipo y se avanzó en la tarea de primera aproximación en Web mediante Flask. Sobre esto último, se creó una visualización del proceso de entrenamiento muy básica por cada iteración.

Se creó un prototipo del algoritmo **CoTraining** sin cumplir con todas sus condiciones que posteriormente se completaron a falta de revisión.

Sobre estos dos algoritmos se propuso la versión 1.0.

Continuando con la Web, se realizó la interfaz general funcional. Incluye:

- Página de Inicio donde seleccionar el algoritmo.
- Página de subida de archivos en formatos ARFF y CSV de los conjuntos de datos
- Páginas correspondientes para SelfTraining y CoTraining: Cada una tiene sus parámetros específicos con la posibilidad de seleccionar si utilizar PCA (Principal Component Analysis) o dos componentes que elija el usuario.
- Página de visualización del algoritmo (su entrenamiento): Se tiene la vista principal que será común a todos los algoritmos (con algunas variaciones en caso necesario) con la posibilidad de avanzar en la visualización (con controles) y barra de progreso. Desde el punto de vista del gráfico los colores están automatizados dependiendo del número de clases, leyenda y etiqueta de ejes.

En el servidor (Flask) a nivel de programación se añadieron los «endpoints» correspondientes (subida, configuración, visualización...) y un control de acceso a las páginas muy básico (por ejemplo, si no se configuró el algoritmo, no se puede visualizar y le redirecciona a la configuración con un mensaje de error)

## Sprint 5

Corresponde con el periodo temporal del 1 al 8 de febrero de 2023.

En la reunión del 1 de febrero se revisó lo realizado en el anterior y se fijó una serie de mejoras/modificaciones y nuevas tareas:

1. Modificación de los algoritmos para trabajar con la convención de «-1s» en el conjunto de datos para los datos no etiquetados. Así el usuario podrá subir un archivo ya **Semi-Supervisado**.
2. Permitir al usuario seleccionar los porcentajes de no etiquetados y de test (para las futuras estadísticas)
3. De la página (no gráfico) visualización de los algoritmos: volver a la configuración, el «feedback» de la iteración actual y el nombre del conjunto de datos utilizado.
4. Del gráfico de la visualización: Diferenciar en el algoritmo CoTraining cuál de los dos clasificadores han etiquetado cada punto y los puntos «etiquetados» en la iteración 0 deben mostrarse de forma diferente.
5. Avanzar con los trabajos relacionados.
6. Avanzar con la documentación teórica y anexos.

El punto 1 ha llevado unas 12 horas de compresión y desarrollo. Esto es debido a que los dos algoritmos implementados hasta ahora debían ser modificados para trabajar con la nueva convención. Además, el problema principal fue (aunque no implementado en este Sprint) dejar preparado una forma de carga del conjunto de datos que permita tratar datos no etiquetados («?» por ejemplo en el caso de ARFF) pues además los algoritmos (su correcto funcionamiento) se han probado con ficheros. También conllevó la creación de un codificador de etiquetas propio para ignorar los no etiquetados en clases categóricas (y no realizar la conversión en esos casos)

El punto 2 volvió a causar bastantes problemas tanto en la ejecución de los algoritmos como en la Web. Hasta el momento, el usuario no seleccionaba los porcentajes de las divisiones. Al incluir esto, los algoritmos ya no se encargan de esta tarea y había que modificar tanto los algoritmos como aquellas rutas de la Web que debían encargarse de esto. Aproximadamente 4 horas.

El punto 3 no resultó demasiado difícil más allá de seguir habituándose a Javascript/HTML. Unas 3 horas.

## **A.3. Estudio de viabilidad**

**Viabilidad económica**

**Viabilidad legal**



## *Apéndice B*

---

# **Especificación de Requisitos**

---

### **B.1. Introducción**

Una muestra de cómo podría ser una tabla de casos de uso:

### **B.2. Objetivos generales**

### **B.3. Catalogo de requisitos**

### **B.4. Especificación de requisitos**

| CU-1                        | Ejemplo de caso de uso  |
|-----------------------------|---|
| <b>Versión</b>              | 1.0   |
| <b>Autor</b>                | Alumno  |
| <b>Requisitos asociados</b> | RF-xx, RF-xx  |
| <b>Descripción</b>          | La descripción del CU   |
| <b>Precondición</b>         | Precondiciones (podría haber más de una)  |
| <b>Acciones</b>             | <ol style="list-style-type: none"> <li>1. Pasos del CU</li> <li>2. Pasos del CU (añadir tantos como sean necesarios)</li> </ol> |
| <b>Postcondición</b>        | Postcondiciones (podría haber más de una)   |
| <b>Excepciones</b>          | Excepciones   |
| <b>Importancia</b>          | Alta o Media o Baja...  |

Tabla B.1: CU-1 Nombre del caso de uso.

## *Apéndice C*

---

# **Especificación de diseño**

---

### **C.1. Introducción**

### **C.2. Diseño de datos**

### **C.3. Diseño procedimental**

### **C.4. Diseño arquitectónico**

### **C.5. Diseño de la Web**

#### **Mockup o Maqueta**

Se presenta el primer Mockup o maqueta de la página Web.

Todas las páginas tendrán una base común en la que aparecerá información general como la Universidad de Burgos (barra superior).

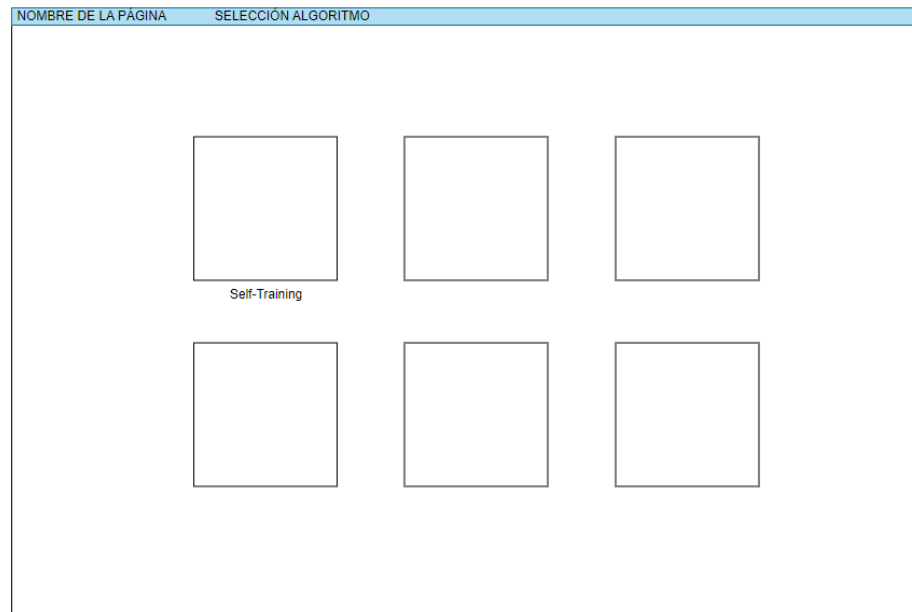



Figura C.1: Página inicial de la Web.

En esta página inicial el usuario podrá seleccionar el algoritmo que desea visualizar. En los cuadrados existirá un logo o imagen representativa del algoritmo junto con su nombre.



NOMBRE DE LA PÁGINA SELF-TRAINING

 SUBIR DATASET ARCHIVO SUBIDO

Precargado de atributos encontrados.  
Usuario selecciona los parámetros

Datasets Locales ▾

Wine  
Breast cancer  
Otros...

Ejecutar

Explicación Self-Training

Pseudocódigo

Figura C.2: Página de configuración del algoritmo.

En esta ventana el usuario podrá subir el conjunto de datos que desee o incluso seleccionar alguno de los almacenados localmente. Además, como los algoritmos tienen parámetros personalizables también habrá elemento para configurarlos.

Antes de iniciar, se muestra una explicación del algoritmo general y su pseudocódigo.

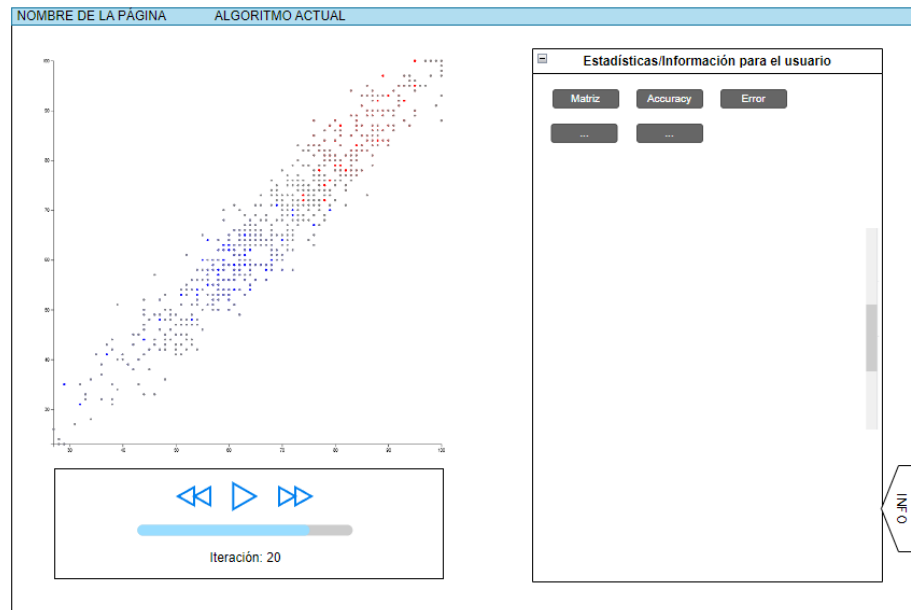


Figura C.3: Página de ejecución del algoritmo.

Mostrará la evolución del entrenamiento de los algoritmos con una vista principal (izquierda) de la clasificación y un compendio de métricas como la precisión o el error en su caso (derecha). Esto último principalmente planteado para ocultar/ver lo que el usuario desee en cada momento.

## *Apéndice D*

---

# **Documentación técnica de programación**

---

- D.1. Introducción
- D.2. Estructura de directorios
- D.3. Manual del programador
- D.4. Compilación, instalación y ejecución del proyecto
- D.5. Pruebas del sistema



## *Apéndice E*

---

# **Documentación de usuario**

---

- E.1. Introducción
- E.2. Requisitos de usuarios
- E.3. Instalación
- E.4. Manual del usuario



---

## **Bibliografía**

---