



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 — 1 de marzo
de 2023

Tutor: Álgvar Arnaiz González
Cotutor: César Ignacio García Osorio

Índice general

Índice general	i
Índice de figuras	iii
Índice de tablas	iv
Apéndice A Plan de Proyecto Software	1
A.1. Introducción	1
A.2. Planificación temporal	1
A.3. Estudio de viabilidad	7
Apéndice B Especificación de Requisitos	9
B.1. Introducción	9
B.2. Objetivos generales	9
B.3. Catalogo de requisitos	9
B.4. Especificación de requisitos	9
Apéndice C Especificación de diseño	11
C.1. Introducción	11
C.2. Diseño de datos	11
C.3. Diseño procedimental	11
C.4. Diseño arquitectónico	11
C.5. Diseño de la Web	11
Apéndice D Documentación técnica de programación	15
D.1. Introducción	15
D.2. Estructura de directorios	15

D.3. Manual del programador	15
D.4. Compilación, instalación y ejecución del proyecto	15
D.5. Pruebas del sistema	15
Apéndice E Documentación de usuario	17
E.1. Introducción	17
E.2. Requisitos de usuarios	17
E.3. Instalación	17
E.4. Manual del usuario	17
Bibliografía	19

Índice de figuras

C.1. Página inicial de la Web.	12
C.2. Página de configuración del algoritmo.	13
C.3. Página de ejecución del algoritmo.	14

Índice de tablas

B.1. CU-1 Nombre del caso de uso.	10
---	----

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 comenzó mediante Sprints de dos 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 fijaron 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. Sobre la página general de la 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 de 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.

El punto 4 requirió unas 10 horas, en un principio se perdió mucho tiempo intentando solucionarlo de una forma que resultó inútil, pero finalmente ahora en el algoritmo se diferencian los datos clasificados por cada uno.

Los trabajos relacionados (no terminados) se realizaron en varios días con un tiempo aproximado de 6 horas.

Sprint 6

Corresponde con el periodo temporal del 8 al 15 de febrero de 2023.

En la reunión del 8 de febrero se revisó lo realizado en el anterior y se comentaron algunas tareas a realizar:

1. En la línea del anterior, los algoritmos deben poder ejecutarse directamente con conjuntos de datos semi-supervisados.
2. Permitir al usuario introducir ese tipo de conjuntos de datos.
3. Realizar alguna visualización de estadísticas.
4. Valor por defecto en las configuraciones.
5. Sobre el gráfico: mejorar la diferenciación de los puntos, información útil en los «tooltips» y colocación leyenda.
6. Avanzar con los trabajos relacionados.
7. Avanzar con la memoria y anexos.

Los puntos 1 y 2 estaban muy avanzados gracias al trabajo adicional del sprint anterior, ya que ya estaba prácticamente implementada la forma en la que detectar datos no etiquetados de forma automática. Unas 5 horas para terminar de implementar, corregir errores sobre la marcha y realizar alguna prueba confeccionando ficheros semi-supervisados.

Al realizar las pruebas anteriores se encontró un error en la visualización provocando que los datos que, por la iteración máxima, no se habían clasificado ni siquiera eran retornadas a la Web. Entre descubrir cómo hacerlo y sus modificaciones se tardó unas 3 horas.

El punto 3 fue el más complicado, pese a que era una idea sencilla, se optó por visualizar la gráfica de la evolución de la precisión. Cada punto del gráfico está unido por una serie de líneas. Este tipo de gráficos (según la documentación) se suelen hacer mediante «paths» o caminos, que son una

única línea, pero como en este caso era necesario no visualizar todo, sino por cada iteración, no se encontró una solución rápida. Unas 6 horas para probar muchas posibilidades hasta encontrar la que funcionó, acoplarla a los controles del paso de iteración e incluir alguna animación.

Adicionalmente se retocó por completo toda la Web mediante los estilos de **Bootstrap** para establecer ya una base vistosa y bonita. Unas 5 horas (la mayor parte del tiempo para probar y adquirir algo de soltura con estos estilos).

Sprint 7

Corresponde con el periodo temporal del 15 al 22 de febrero de 2023.

Puntos a desarrollar:

1. Implementación Democratic Co-Learning.
2. Profiling (tiempos de ejecución).
3. Estadísticas en Web.
4. Test de las implementaciones.
5. Avanzar con la memoria y anexos.

La implementación del algoritmo Democratic Co-Learning supuso unas 14 horas divididas en varios días. Al principio se dedicó un tiempo para leer el artículo en el que se presentaba su implementación en forma de pseudocódigo junto con sus explicaciones teóricas. La realidad es que en primera instancia parecía algo fácil de realizar y entender, pero una vez comenzada la implementación se encontraban muchas alternativas a la hora de resolverlo. Además, pese a que en el artículo estaba bien explicado, el formato de pseudocódigo (en el archivo encontrado) las indentaciones eran incorrectas y se perdió mucho tiempo comprobando si era una interpretación errónea o si realmente era un fallo.

Se realizaron algunas pruebas de rendimiento para comprobar si los algoritmo tardaban demasiado con conjuntos de datos muy grandes (5 000 instancias). Se observó que, dada la configuración que se tenía, tardaba alrededor de 40-50 segundos en terminar la ejecución. Es por esto que para este Sprint se añadió la tarea de hacer un pequeño estudio dedicado al medir los tiempos de ejecución para ver qué se podía optimizar. Este proceso fue

de unas 2 horas y el resultado es que el código implementado no afectaba mucho, eran los propios algoritmos de entrenamiento de los clasificadores de Scikit-Learn los que tardaban tanto. Por ejemplo, para un estimador gaussiano el tiempo se reducía drásticamente.

Para el caso de las estadísticas, se modificaron un poco las plantillas y la generación de sus gráficas para incluir más y revisarlas en la reunión. Unas 2 horas.

Los test son una parte importante para validar que el comportamiento que se espera de la implementación sea el correcto. Se realizaron unos casos de pruebas sobre las utilidades que se usan a lo largo de todo el proyecto con la intención de encontrar errores (todo esto sin ver cuál es el resultado y replicarlo en los casos, sino realizar los casos basándose en lo que se espera de esas utilidades). Se tardó unas 4 horas en realizar todos los tests.

Sprint 8

Corresponde con el periodo temporal del 22 de febrero al 1 de marzo de 2023. Además, aprovechando la herramienta Zenhub, se ha modificado la duración de los Sprints también en ella para poder extraer los gráficos del trabajo realizado.

Puntos a desarrollar:

1. Intervalo de confianza en Democratic Co-Learning.
2. Control re-etiquetado en Democratic Co-Learning
3. Correcciones sobre memoria y anexos.
4. Gráfico de estadísticas unificado.
5. Internacionalización Web.
6. Visualización principal de Democratic Co-Learning en Web

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
Versión	1.0
Autor	Alumno
Requisitos asociados	RF-xx, RF-xx
Descripción	La descripción del CU
Precondición	Precondiciones (podría haber más de una)
Acciones	<ol style="list-style-type: none"> 1. Pasos del CU 2. Pasos del CU (añadir tantos como sean necesarios)
Postcondición	Postcondiciones (podría haber más de una)
Excepciones	Excepciones
Importancia	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 que se comentó de la página Web. Todas las páginas tendrán una base común en la que aparecerá información general como la Universidad de Burgos (barra superior).

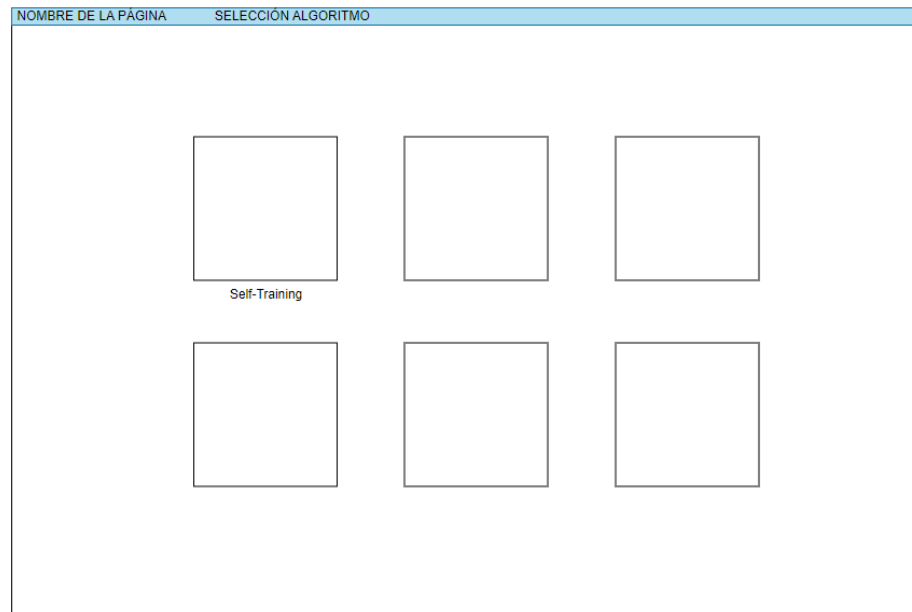


Figura C.1: Página inicial de la Web.

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.

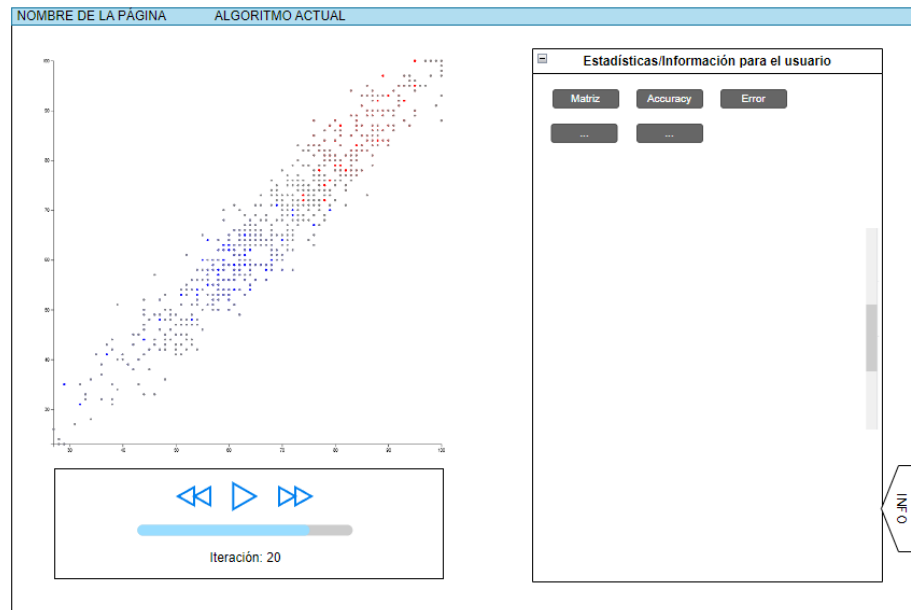


Figura C.3: Página de ejecución del algoritmo.

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
