



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería en Informática



TFG del Grado en Ingeniería Informática  
título del TFG



Presentado por Nombre del alumno  
en Universidad de Burgos — 2 de noviembre de 2017  
Tutor: nombre tutor





UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería en Informática



D. nombre tutor, profesor del departamento de nombre departamento, área de nombre área.

Expone:

Que el alumno D. Nombre del alumno, con DNI dni, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado título de TFG.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 2 de noviembre de 2017

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. nombre tutor

D. nombre co-tutor



## **Resumen**

En este primer apartado se hace una **breve** presentación del tema que se aborda en el proyecto.

## **Descriptores**

Palabras separadas por comas que identifiquen el contenido del proyecto Ej: servidor web, buscador de vuelos, android . . .

### **Abstract**

A **brief** presentation of the topic addressed in the project.

### **Keywords**

keywords separated by commas.

---

# Índice general

---

Índice general	III
Índice de figuras	IV
Índice de tablas	V
Introducción	1
Objetivos del proyecto	2
2.1. Objetivos . . . . .	2
Conceptos teóricos	3
3.1. Secciones . . . . .	4
3.2. Referencias . . . . .	4
3.3. Imágenes . . . . .	4
3.4. Listas de items . . . . .	5
3.5. Tablas . . . . .	6
Técnicas y herramientas	7
4.1. GitHub: . . . . .	7
4.2. Spyder: . . . . .	8
4.3. LaTeX: . . . . .	8
4.4. Jupyter notebook: . . . . .	8
Aspectos relevantes del desarrollo del proyecto	9
Trabajos relacionados	10
Conclusiones y Líneas de trabajo futuras	11
Bibliografía	12

---

# Índice de figuras

---

3.1. Autómata para una expresión vacía . . . . .	5
--	---



---

# Índice de tablas

---

3.1. Herramientas y tecnologías utilizadas en cada parte del proyecto .	6
---	---

---

# Introducción

---

Este proyecto trata sobre implementar algoritmos de clasificación multi-label mediante ensembles, son clasificadores que combinan predicciones de otros clasificadores. Los clasificadores combinados en un ensemble son conocidos como clasificadores base. La precisión global de los ensembles necesita que los clasificadores base predigan correctamente la clase de las mismas instancias. Tienen que ser diferentes para complementarse entre ellos. ¿Cómo puede un ensemble de clasificadores base que han sido generados por el mismo algoritmo tener distintas salidas, si las entradas son las mismas? Esto se ha conseguido en ensembles que usan distintas estrategias, los ensembles se suelen basar en la modificación del conjunto de datos de entrenamiento de clasificadores base.

---

# Objetivos del proyecto

---

Este apartado explica de forma precisa y concisa cuales son los objetivos que se persiguen con la realización del proyecto. Se puede distinguir entre los objetivos marcados por los requisitos del software a construir y los objetivos de carácter técnico que plantea a la hora de llevar a la práctica el proyecto.

## 2.1. Objetivos

A continuación se mostrara el esquema con todos los puntos a tratar en este proyecto.

- Implementar un algoritmo en Scikit-learn:
  - Crear los métodos fit para entrenar y predict para predecir.
  - Crear el método predict-prob para predecir probabilidades.
  - Probar que la clase funciona correctamente.
- Crear un notebook para mostrar:
  - Llamar a la clase para mostrar los resultados en notebooks de jupyter.
  - Crear una semilla para pasar a la clase.
  - Mostrar en un árbol la clasificación del conjunto.

---

## Conceptos teóricos

---

- Scikit-learn: Es una librería de aprendizaje de software libre en el lenguaje de programación de Python. Es una herramienta de las más utilizadas para la minería de datos y el análisis de datos. Esta basada en el aprendizaje automático, para ello se consideran un conjunto de  $n$  muestras y se intenta predecir las propiedades de los datos desconocidos. Podemos separar los problemas de aprendizaje en dos:
- Minería de Datos: Es un conjunto de reglas mediante las cuales se analizan grandes volúmenes de datos, la finalidad de esto es descubrir unos patrones, una similitud o una propensión que expliquen el comportamiento de los datos. Para hacer esto utiliza los métodos de la inteligencia artificial, estadística y redes neuronales. El objetivo del proceso de minería de datos consiste en extraer información de un conjunto de datos, luego se interpreta esta información para un uso posterior. En general el proceso de la minería de datos consta de 4 etapas:
  - Determinación de los objetivos: Se tratan los objetivos que quiere conseguir el cliente bajo un asesor especialista en minería de datos.
  - Preprocesamiento de los datos: Es la etapa que más tiempo se tarda en realizar en el proceso. Se seleccionan, limpian, enriquecen, reducen y transforman las bases de datos.
  - Determinación del modelo: Se lleva a cabo un estudio estadístico de los datos, más tarde se hace una visualización gráfica para una primera aproximación. Según los objetivos que se habían propuesto se pueden usar diferentes algoritmos de la Inteligencia Artificial.
  - Análisis de los resultados: Se comprueban si los datos obtenidos tienen coherencia, después se comparan con los obtenidos en los estudios estadísticos y la visualización gráfica. El cliente es el que ve si los datos le aportan nuevo conocimiento que le permita considerar sus decisiones.

-Aprendizaje supervisado: En el que los datos vienen con atributos adicionales que queremos predecir. Dos de las tareas mas comunes son la clasificación y la regresión. En las de clasificación el programa debe aprender a predecir en que categoría o clase irán los nuevos datos según las nuevas observaciones, como sería predecir si el precio de una acción bajara o subirá. En lo de regresión el programa debe predecir el valor de una variable de respuesta continua, como sería predecir las ventas de un nuevo producto.

-Aprendizaje no supervisado: Consiste en agrupar observaciones relaciones, dentro de los datos del entrenamiento. Clustering es la que más se utiliza para explorar un conjunto de datos.

Algunos conceptos teóricos de L<sup>A</sup>T<sub>E</sub>X<sup>1</sup>.

### 3.1. Secciones

Las secciones se incluyen con el comando section.

#### Subsecciones

Además de secciones tenemos subsecciones.

#### Subsubsecciones

Y subsecciones.

### 3.2. Referencias

Las referencias se incluyen en el texto usando cite [2]. Para citar webs, artículos o libros [1].

### 3.3. Imágenes

Se pueden incluir imágenes con los comandos standard de L<sup>A</sup>T<sub>E</sub>X, pero esta plantilla dispone de comandos propios como por ejemplo el siguiente:

---

<sup>1</sup>Créditos a los proyectos de Álvaro López Cantero: Configurador de Presupuestos y Roberto Izquierdo Amo: PLQuiz



Figura 3.1: Autómata para una expresión vacía

### 3.4. Listas de ítems

Existen tres posibilidades:

- primer ítem.
- segundo ítem.

1. primer ítem.
2. segundo ítem.

Herramientas	App	AngularJS	API REST	BD	Memoria
HTML5		X			
CSS3		X			
BOOTSTRAP		X			
JavaScript		X			
AngularJS		X			
Bower		X			
PHP			X		
Karma + Jasmine		X			
Slim framework			X		
Idiorm			X		
Composer			X		
JSON		X	X		
PhpStorm		X	X		
MySQL				X	
PhpMyAdmin				X	
Git + BitBucket		X	X	X	X
MikTeX					X
TeXMaker					X
Astah					X
Balsamiq Mockups		X			
VersionOne		X	X	X	X

Tabla 3.1: Herramientas y tecnologías utilizadas en cada parte del proyecto

**Primer ítem** más información sobre el primer ítem.

**Segundo ítem** más información sobre el segundo ítem.

■

### 3.5. Tablas

Igualmente se pueden usar los comandos específicos de  $\text{\LaTeX}$  o bien usar alguno de los comandos de la plantilla.

---

# Técnicas y herramientas

---

Esta parte de la memoria tiene como objetivo presentar las técnicas metodológicas y las herramientas de desarrollo que se han utilizado para llevar a cabo el proyecto. Si se han estudiado diferentes alternativas de metodologías, herramientas, bibliotecas se puede hacer un resumen de los aspectos más destacados de cada alternativa, incluyendo comparativas entre las distintas opciones y una justificación de las elecciones realizadas. No se pretende que este apartado se convierta en un capítulo de un libro dedicado a cada una de las alternativas, sino comentar los aspectos más destacados de cada opción, con un repaso somero a los fundamentos esenciales y referencias bibliográficas para que el lector pueda ampliar su conocimiento sobre el tema.

## 4.1. GitHub:

Es una plataforma para alojar proyectos utiliza git como sistema de control de versiones. Se organiza por tareas(milestones e issues). Utiliza el framework Ruby on Rails.

Podemos acceder a él a través del siguiente enlace: <https://github.com/>

### **Ventajas:**

- Es uno de los repositorios mas usados, por lo que es fácil encontrar información en Internet para resolver cualquier duda.
- El código es público por lo que cualquiera puede proponer cambios en el mismo, seguirte y ver el proyecto.
- Las distintas versiones del código están alojadas en la nube por lo que si perdemos el contenido de nuestro ordenador, podremos recuperarlo.

### **Desventajas:**



- También puedes tener proyector privados pero para ello tienes que utilizar una cuenta de pago.

## 4.2. Spyder:

Es un entorno de desarrollo interactivo para el lenguaje de Python, es de código abierto. Tiene funciones avanzadas de edición, pruebas interactivas, depuración e introspección. También es un entorno informático numérico y tiene diversas bibliotecas que podemos utilizar.

Podemos acceder a él a través del siguiente enlace: <http://pythonhosted.org/spyder/>

## 4.3. LaTeX:

Se usa para la creación de documentos que necesiten una alta calidad tipográfica, como puede ser en artículos o libros científicos.

### Ventajas:

- Es software libre, por lo que no requiere ningún coste.
- No te tienes preocupar por el diseño, ya que la herramienta se encarga de ello.

### Desventajas:

- Si eres principiante necesitas un tiempo de aprendizaje para saber como funciona.

Podemos acceder a él a través del siguiente enlace: <https://www.latex-project.org/>

## 4.4. Jupyter notebook:

Es una aplicación web de código abierto, con él podemos crear documentos, que nos permita visualizar los resultados al ejecutar nuestro código, ya sean imágenes, árboles...que otros entornos de desarrollo (como Spyder mencionado anteriormente) no nos permiten esto. Soporta más lenguajes pero nosotros lo usaremos para el lenguaje de Python.

Podemos acceder a él a través del siguiente enlace: <http://jupyter.org/>

---

## Aspectos relevantes del desarrollo del proyecto

---

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros<sup>3</sup>, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.

---

## Trabajos relacionados

---

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final grado no parece obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.

---

## **Conclusiones y Líneas de trabajo futuras**

---

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

---

# Bibliografía

---

- [1] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
- [2] Wikipedia. Latex — wikipedia, la enciclopedia libre, 2015. [Internet; descargado 30-septiembre-2015].