



**Universidad de Jaén**

*Escuela Politécnica Superior de Jaén*

# **Reproducción, evaluación y adaptación de modelos de seguimiento visual para escenarios Anti-UAV con imágenes térmicas**

**Autor: Álvaro Cobo Mayoral**

**Grado: Ingeniería en Informática**

**Directores: Antonio Jesús Rivera Rivas y Pedro González García**  
Departamento del director: Informática

Fecha: 3/11/2025

Licencia CC



CREA





UNIVERSIDAD DE JAÉN

D./D<sup>a</sup> Antonio Jesús Rivera Rivas y D./D<sup>a</sup> Pedro González García, tutor(es) del Trabajo Fin de Grado titulado: **Reproducción, evaluación y adaptación de modelos de seguimiento visual para escenarios Anti-UAV con imágenes térmicas**, que presenta Álvaro Cobo Mayoral, autoriza(n) su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, 3/11/2025



## **Agradecimientos**

Incluir aquí los agradecimientos que se deseen, dedicatoria y cualquier otro texto de carácter personal.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.



# Tabla de contenidos

<b>1. INTRODUCCIÓN</b>	<b>1</b>
1.1. Motivación	1
1.1.1. Orígenes de la tecnología UAV	1
1.1.2. Riesgos y amenazas emergentes	1
1.2. Seguimiento de Objetos en imágenes térmicas infrarrojas	2
1.2.1. Seguimiento de objetos individuales (SOT)	2
1.2.2. Imágenes térmicas en el espectro infrarrojo	3
1.2.3. Capítulos	3
1.2.4. Secciones	4
1.2.5. Subsecciones y subsubsecciones	4
1.2.6. Párrafos de texto	4
1.2.7. Referencias a capítulos, secciones y subsecciones	5
1.3. Texto en negrita, cursiva y monoespacio	5
1.4. Cómo insertar notas	6
1.5. Cómo insertar listas numeradas y sin numerar	6
1.6. Cómo insertar figuras	7
1.6.1. Aspectos importantes al usar figuras	7
1.6.2. Cómo almacenar las figuras	8
1.6.3. Cómo insertar las figuras en el documento	8
1.7. Cómo insertar tablas	9
1.7.1. Aspectos importantes al usar tablas	9
1.7.2. Cómo almacenar las tablas	10
1.7.3. Cómo insertar las tablas en el documento	10
1.8. Cómo insertar fórmulas/ecuaciones	12
1.9. Cómo insertar algoritmos	13
1.10. Cómo insertar código	14
1.11. Cómo citar textos y otros materiales	15
1.11.1. Citas bibliográficas	15
1.11.2. Citas textuales	18
1.12. Cómo introducir ciertos caracteres en $\text{\LaTeX}$	19

1.13. Cómo obtener esta plantilla . . . . .	19
1.14. Cómo configurar la plantilla con nuestros datos . . . . .	19
1.15. En cuanto al contenido de los capítulos . . . . .	20
1.15.1. TFG de tipo <i>Trabajo teórico/experimental</i> . . . . .	20
1.15.2. TFG de tipo <i>Proyecto de ingeniería</i> . . . . .	22
1.15.3. Otros archivos del proyecto . . . . .	24
1.16. Consejos sobre redacción . . . . .	24
<b>2. ANTECEDENTES</b>	<b>27</b>
<b>3. OBJETIVOS</b>	<b>29</b>
<b>4. MATERIALES Y MÉTODOS</b>	<b>31</b>
<b>5. RESULTADOS</b>	<b>33</b>
<b>6. CONCLUSIONES</b>	<b>35</b>
<b>Bibliografía</b>	<b>I</b>



# Lista de figuras

1.1. Logo oficial de la Universidad de Jaén. . . . .	9
--	---



# Lista de tablas

1.1. Requisitos hardware del ordenador en el que se instalará el sistema . . . . .	12
--	----



# Lista de algoritmos

1. Título del algoritmo descrito . . . . . 13



# Lista de listados de código

1.1. Creación de una lista no numerada con dos elementos . . . . .	7
1.2. Insercción de una figura en el documento . . . . .	8
1.3. Creación de una tabla . . . . .	11
1.4. Definición de una ecuación . . . . .	13
1.5. Descripción de un algoritmo . . . . .	14
1.6. Fuente LaTeX que genera el listado en lenguaje SQL . . . . .	14
1.7. Creación de una tabla con SQL . . . . .	15
1.8. Entradas BibTex de distintos tipos . . . . .	17
1.9. Párrafo en el que se citan varios trabajos . . . . .	17
1.10. Variables a establecer en <code>main.tex</code> . . . . .	19





# Capítulo 1

## INTRODUCCIÓN

### 1.1. Motivación

#### 1.1.1. Orígenes de la tecnología UAV

Desde hace dos siglos, el espacio aéreo ha dejado de ser una frontera inalcanzable para integrarse plenamente, junto al dominio terrestre y marítimo, como un área de proyección humana. En la actualidad, el cielo se consolida como un dominio estratégico fundamental para el transporte, las comunicaciones globales y la seguridad internacional. Sin embargo, y pese a lo antiintuitivo que pueda parecer, el concepto de aeronave no tripulada con fines estratégico-militares se remonta hasta comienzos del siglo pasado, coincidiendo con el desarrollo de la primera guerra mundial.

Desde el comienzo, demostraron ser prometedoras herramientas de reconocimiento aéreo en las pruebas de vuelo, pero no fue hasta el estallido de la guerra de Vietnam (1946-1954) cuando se desplegaron a gran escala y comenzaron a usarse con diversos fines tales como actuar como señuelos en combate o lanzar misiles contra objetivos fijos entre otros.

Tras este conflicto, muchos países comenzaron a invertir en su desarrollo, exploración y avance tecnológico, con nuevos modelos que mejoraban la resistencia y la capacidad de elevarse a mayor altitud. Actualmente, los drones mantienen muchos fines y aplicaciones reales, pero sin duda, su uso más conocido y relevante es el militar, en tareas de vigilancia, reconocimiento y ataques dirigidos.

#### 1.1.2. Riesgos y amenazas emergentes

A partir de noviembre de 2013 los microdrones y los UAVs fueron aprobados para su integración en el espacio aéreo de baja altitud (LAA) por la Administración Federal de Aviación (FAA). Esto ha provocado un crecimiento muy significativo en el número de drones registrados (más de 670.000 en EE. UU. en 2016).Consecuentemente, surge una necesidad urgente de generar sistemas de legislación y regulación de estos vehículos a nivel nacional e

internacional.

Sin embargo, esto deriva en varios problemas, como la dificultad de países no desarrollados para implementar leyes y sistemas de registro tan estrictos que rijan el uso de drones. Además, el tan elevado número de estos vehículos suele provocar la falta de personal de cumplimiento activo para vigilar estas leyes, así como sistemas de detección operativos y eficientes, lo cual suele dejar a la policía o fuerzas del orden sin suficientes recursos para lidiar con estas amenazas.

El potencial de estas tecnologías para causar daños a la seguridad pública o nacional y para cometer delitos graves (tales como la violación de la privacidad, el transporte de sustancias ilegales, colisiones inadvertidas, el despliegue de armas explosivas o agentes químicos) es enorme. El ejemplo más recurrente de el daño que estas tecnologías pueden causar es el de los peligros de colisión, donde un artículo del IEEE en 2011 (citar) demostró que un dron de 2kg retiene la energía cinética equivalente a un proyectil antiaéreo de 20 mm (casi el doble de grande que uno de calibre 0,50).

## **1.2. Seguimiento de Objetos en imágenes térmicas infrarrojas**

### **1.2.1. Seguimiento de objetos individuales (SOT)**

Dadas las numerosas problemáticas mencionadas en la anterior sección, surge la necesidad lógica de desarrollar sistemas automáticos capaces de detectar y seguir estos vehículos. Al eliminar el factor del error humano y las limitaciones ópticas, estos sistemas permiten mantener un espacio aéreo seguro, tanto en ámbitos de seguridad nacional como en la protección de la privacidad y el derecho a la intimidad, y recopilar pruebas forenses (estos sistemas pueden triangular la señal del dron para localizar al operador en caso de infracciones legales).

Es en este contexto donde surge el concepto SOT (Single Object Tracking). El seguimiento o trackeo de objetivos individuales es una tarea fundamental en el campo de la visión por computador o visión artificial (computer vision). Los sistemas SOT tienen como función rastrear un objeto concreto a lo largo de un video. El objetivo se especifica en el primer fotograma y debe ser reconocido y seguido en los fotogramas sucesivos. Esta técnica, junto con el seguimiento multi-objetivo (MOT), son las técnicas fundamentales en las aplicaciones de sistemas anti-UAVs.

Sin embargo, como en todos los campos de investigación; los sistemas y algoritmos SOT se enfrentan a numerosos desafíos y limitaciones, como los cambios repentinos de iluminación entre fotogramas, el ruido de fondo de las imágenes, la oclusión del objetivo y la escala del mismo entre otros. Debido a la presencia de estos factores en situaciones reales,

los sistemas SOT deben cumplir tres características generales:

- **Adaptabilidad:** El sistema debe ser capaz de rastrear el objetivo incluso en condiciones adversas como ruido de fondo, oclusión o cambios de iluminación.
- **Robustez:** Se debe garantizar que además de las alteraciones de entorno, el sistema debe adaptarse a cambios como los movimientos complejos y repentinos del objetivo, Algo muy común en los microdrones y UAVs. Para lograr esto, el sistema debe ser capaz de detectar y rastrear las características aparentes del objetivo.
- **Eficiencia:** La mayoría de sistemas SOT se enfrenta a situaciones de rastreo en tiempo real, por lo que estos deben tener una alta velocidad general de procesamiento, lo que requiere implementar algoritmos de alto rendimiento.

### 1.2.2. Imágenes térmicas en el espectro infrarrojo

Uno de los principales retos a los que se enfrentan los sistemas de seguimiento de UAVs es la condición del entorno. Un buen sistema SOT debe poder detectar y seguir objetivos en cualquier momento del día y bajo cualquier condición adversa (temperatura, humedad, temporal...), lo cual es especialmente complicado en las horas nocturnas o de poca luz natural, y en entornos con niebla, lluvia, rachas fuertes de viento y demás fenómenos ambientales. Estos factores no solo reducen a menudo la calidad de la imagen, sino que además introducen ruido y variaciones en la firma del objetivo. Es por esto que la tecnología de imágenes infrarrojas comenzó a ser muy relevante en este campo.

La tecnología de procesamiento de imágenes térmicas en el espectro infrarrojo (TIR) consiste en la reconstrucción de imágenes a través de la radiación infrarroja emitida por los objetos de la escena. Una de las principales ventajas de este proceso es su capacidad para no interferir directamente con el objetivo, ya que la radiación electromagnética es una propiedad física inherente a cualquier cuerpo con una temperatura por encima del cero absoluto. Al limitarse a recibir dicha señal sin emitir energía externa, los detectores TIR se clasifican como sistemas de detección pasivos, lo que los hace ideales para aplicaciones militares y de vigilancia.

Pese a todas sus ventajas, el proceso de obtención de estas imágenes requiere una serie de etapas de procesamiento debido al bajo contraste y a la escasa resolución de los detalles de la imagen. Estas etapas de procesamiento suelen llevarse a cabo a través técnicas de visión artificial (computer vision) y aprendizaje profundo (deep learning).

### 1.2.3. Capítulos

La memoria se estructura en varios capítulos, en principio esta plantilla tiene tantos como apartados indica la normativa de la EPSJ, si bien existe cierta flexibilidad a la hora de agregar otros que se consideren necesarios.

El fuente  $\text{\LaTeX}$  para cada capítulo se alojará en un archivo independiente, con extensión `.tex`, que será incluido en el documento principal con el comando `\input`. La primera línea del capítulo será:

```
\chapter{TÍTULO DEL CAPÍTULO}
```

Observa que el título se escribe completamente en mayúsculas. La numeración de los capítulos es automática, por lo que no debes preocuparte por este detalle.

### 1.2.4. Secciones

Un capítulo se dividirá en secciones, cada una de las cuales se inicia con el siguiente comando:

```
\section{Título de la sección}
```

Es recomendable planificar las secciones en que se dividirá un capítulo antes de comenzar a escribirlo. Debemos pensar qué queremos contar en el capítulo y cómo estructurar ese contenido en partes cohesionadas, cada una de las cuales daría lugar a una sección.

### 1.2.5. Subsecciones y subsubsecciones

Las secciones de cierta extensión se dividirán en subsecciones. De igual forma, una subsección con mucho contenido se dividiría en subsecciones. Los comandos para generar estos niveles son:

```
\subsection{Título de la subsección}
```

```
\subsubsection{Título de la subsubsección}
```

Aunque obviamente es algo subjetivo, en general no es deseable tener varias páginas seguidas de texto sin ninguna división que permita tener un punto de referencia.

### 1.2.6. Párrafos de texto

Tras los comandos `\chapter`, `\section`, `\subsection` y `\subsubsection` dispondremos los párrafos de texto con las descripciones y explicaciones que procedan. Es recomendable tener al menos un párrafo de texto entre dos comandos sucesivos de los anteriores, evitando así tener el inicio de una sección justo tras el inicio del capítulo, de una subsección inmediatamente tras la sección, etc.

Los párrafos de texto en  $\text{\LaTeX}$  se separan unos de otros con una línea en blanco. El contenido de un párrafo se escribe todo seguido, sin pulsar nunca **Intro/Enter**. Al compilar se ajustará el texto automáticamente, introduciendo guiones donde sea necesario.

## 1.2.7. Referencias a capítulos, secciones y subsecciones

Al escribir un documento extenso, como es el caso de una memoria de TFG, es habitual la necesidad de hacer referencia a partes del mismo (capítulo, sección, subsección) desde ciertos puntos del texto. Por ejemplo, en la Sección 1.2.3 se describe cómo dividir el documento principal en capítulos.

Crear estas referencias en  $\text{\LaTeX}$  es muy sencillo y se compone de dos pasos:

1. Colocar una etiqueta tras la llave de cierre correspondiente al comando `\chapter`, `\section`, `\subsection` o `\subsubsection`, según el caso, usando para ello el comando `\label{etiqueta}`.
2. Desde cada punto del texto en que necesitemos hacer referencia a ese capítulo, sección o subsección usaríamos el comando `\ref{etiqueta}`, que insertará el número que corresponda.

En este propio fuente puedes ver ejemplos de uso de dichos comandos. Además de `\ref`, que inserta el número de capítulo, sección, subsección o subsubsección, también se pueden usar los comandos `\nameref` para obtener el título y `\pageref` para obtener la página. Por ejemplo, el comando<sup>1</sup>:

En la Sección~\ref{Sec.Capitulos}~\nameref{Sec.Capitulos} (véase la pág.~\pageref{Sec.Capitulos}) se explica cómo estructurar los capítulos.

Produce el siguiente resultado:

*En la Sección 1.2.3 Capítulos (véase la pág. 3) se explica cómo estructurar los capítulos.*

## 1.3. Texto en negrita, cursiva y monoespacio

Los estilos tipográficos de las distintas partes de la memoria, como son los títulos de capítulos, secciones o los propios párrafos de texto, están predeterminados. En el contenido podemos emplear esencialmente tres variantes:

- **Negrita:** debe usarse de forma puntual para **destacar algo importante**, evitando abusar de ello ya que si hay mucho texto en negrita el efecto tendrá poca utilidad. Se usa el comando `\textbf{texto en negrita}`.
- *Cursiva:* se emplea generalmente para introducir términos en otro idioma, por ejemplo GPU (*Graphics Processing Unit*), y esporádicamente con otros fines. Se usa el comando `\textit{texto en cursiva}`.

---

<sup>1</sup>El carácter ~ en  $\text{\LaTeX}$  actúa como un espacio irrompible. Permite garantizar que las partes dispuestas a izquierda y derecha no se separarán, saltando juntas a la siguiente línea de texto en caso de que no hubiese espacio suficiente en la actual

- **Monoespacio:** se usa para diferenciar nombres de variables, funciones, archivos y, en general, todo lo relacionado con código, por ejemplo: la función `printf()` de C. Se usa el comando `\texttt{texto en monoespacio}`.

Aunque  $\text{\LaTeX}$  nos permite trabajar con colores (`\textcolor{color}{texto}`), diferentes tamaños de fuente (`\normalsize`, `\small`, `\footnotesize`, etc.), recuadros con fondo de color, etc., en la normativa de TFG de la EPSJ no se especifica que pueda hacerse uso de los mismos, por lo que se desaconseja hacerlo.

## 1.4. Cómo insertar notas

Las notas a pie de página<sup>2</sup> son un recurso útil para aclarar algo o facilitar información complementaria sin romper el hilo de lo que está tratándose en el texto. Al igual que el texto en negrita, es un recurso del que tampoco debe abusarse, empleándose únicamente cuando es estrictamente necesario.

Para incluir una nota al pie nos serviremos del comando `\footnote{Texto}`, colocándolo allí donde queremos que aparezca el número de la nota como superíndice, tal y como se aprecia en el fuente  $\text{\LaTeX}$  de este mismo texto.

En casos excepcionales, cuando ya no queda espacio disponible en la página actual, una nota al pie puede desplazarse a la página siguiente de donde se hace referencia a ella.

## 1.5. Cómo insertar listas numeradas y sin numerar

Un recurso muy habitual al redactar una memoria son las listas, tanto numeradas como sin numerar. Por su disposición en el texto, con el sangrado y las marcas que preceden cada elemento, es recomendable usar listas para cualquier enumeración de elementos o descripción de procedimientos compuestos de múltiples pasos. La lista numerada se debería usar siempre y cuando el orden sea importante, en caso contrario es preferible una lista sin numerar.

Las listas, al igual que otros elementos en  $\text{\LaTeX}$ , puede ocupar múltiples párrafos de texto, incluso extendiéndose por varias páginas, de ahí la necesidad de marcar su inicio y su fin, definiendo lo que se denomina **un ámbito** o *environment*. Los ámbitos siempre se inician con el comando `\begin{}` y se finalizan con el comando `\end{}`. El tipo de ámbito, y por tanto su contenido, viene determinado por el nombre que dispongamos entre las llaves.

Para el caso de las listas usaremos dos ámbitos distintos:

- `itemize`: para generar una lista no numerada
- `enumerate`: para producir una lista numerada

---

<sup>2</sup>Estas notas se numeran automáticamente dentro de cada capítulo de la memoria.

En el interior del ámbito usaremos el comando `\item` para ir introduciendo los elementos que forman la lista, por ejemplo:

```
1 \begin{itemize}
2   \item \texttt{itemize}: para generar una lista no numerada
3   \item \texttt{enumerate}: para producir una lista numerada
4 \end{itemize}
```

Listado 1.1: Creación de una lista no numerada con dos elementos

Las listas pueden anidarse, es decir, podemos tener una lista dentro de otra, independientemente de cuál sea el tipo de cada una. El sangrado, tipo de boliche<sup>3</sup> y esquema de numeración se ajustarán de forma automática.

## 1.6. Cómo insertar figuras

A lo largo de una memoria de TFG es preciso incluir distintos tipos de figuras: diagramas que describen cómo funciona un sistema, ilustraciones y gráficas que representan o evalúan unos resultados, capturas de pantalla de un programa, etc.

### 1.6.1. Aspectos importantes al usar figuras

Las normas a tener presentes a la hora de introducir figuras en la memoria son las indicadas a continuación:

- Todas las figuras habrán de estar numeradas, usando una numeración consecutiva para toda la memoria o bien relativa a cada capítulo.
- Todas las figuras han de contar con un pie de figura que describa claramente qué representa. No ocurre nada si el pie de figura es extenso y ocupa varias líneas.
- Todas las figuras han de estar referenciadas desde el texto de la memoria, es decir, en algún punto de los párrafos previos o posteriores ha de aparecer una referencia del tipo «*como se aprecia en la Figura N*».
- **Importante:** todas las figuras deberán indicar su procedencia, de no indicarse se entenderá que son de elaboración propia.

Trata siempre de que las figuras tengan una calidad suficiente, que en el documento resultante no se vean borrosas o pixeladas, ya que causan un mal efecto.

---

<sup>3</sup>La marca que aparece al inicio de cada elemento de una lista no numerada

### 1.6.2. Cómo almacenar las figuras

En ocasiones las figuras a usar serán capturas de pantalla, en otras gráficas producidas con algún programa y otras imágenes procedentes de alguna fuente. Siempre que sea posible, por ejemplo para gráficas propias generadas a partir de datos o diagramas confeccionados con alguna aplicación, se recomienda almacenar la imagen en un formato vectorial, como puede ser EPS o PDF, ya que al insertarse en el documento generalmente conservan una mejor calidad.

Para imágenes no vectoriales puedes usar los formatos PNG y JPEG. Dependiendo de la resolución de la pantalla en que se haga la captura, la imagen tendrán mayor o menor calidad. En caso necesario, se puede usar la utilidad `convert` del software ImageMagick ([legacy.imagemagick.org](http://legacy.imagemagick.org)) para incrementar el número de puntos por pulgada, de manera que al insertar la imagen en el PDF mejore su calidad.

Se recomienda dar una denominación adecuada a los archivos donde se guardan las figuras, de forma que sea fácil de recordar para luego introducirlas en el fuente L<sup>A</sup>T<sub>E</sub>X. Asimismo, es recomendable llevar las imágenes a una carpeta aparte, ya sea general para todas las figuras o bien por capítulo.

### 1.6.3. Cómo insertar las figuras en el documento

El comando para incluir un archivo de tipo gráfico, ya sea vectorial o no, en un documento es `\includegraphics[]{}{}`. Entre los corchetes se usarán habitualmente los parámetros `width` o `height` para establecer el ancho o alto que queremos dar a la figura. Lo más recomendable es usar dimensiones relativas al espacio disponible, por ejemplo: `[width=0.5\textwidth]` para conseguir que la figura ocupe la mitad del ancho de página y se mantenga la proporción alto/ancho. Dentro de las llaves se indicará el nombre del archivo que contiene la imagen.

Dado que aparte de la imagen en sí la figura ha de contar también con un pie o título, definido con el comando `\caption`, y además es habitual asociar una etiqueta a la figura para así poder hacer referencia a ella desde el texto, usando el comando `\label` que se explicó anteriormente, es preciso agrupar todos estos elementos dentro de un ámbito de tipo `figure`. Por ejemplo, el código del Listado 1.2 daría lugar a la inserción de la Figura 1.1. El comando `\centering`, introducido dentro del ámbito `figure`, hace que la figura aparezca centrada en lugar de alineada a la izquierda.

```
1 \begin{figure}[ht!]  
2   \centering  
3   \includegraphics[width=0.4\textwidth]{imagenes/uja.jpg}  
4   \caption{Logo oficial de la Universidad de Jaen.}  
5   \label{Fig.LogoUJA}  
6 \end{figure}
```

Listado 1.2: Inserción de una figura en el documento





Figura 1.1: Logo oficial de la Universidad de Jaén.

Las referencias desde el texto a la figura se generarían según lo explicado en la subsección [1.2.7 Referencias a capítulos, secciones y subsecciones](#).

## 1.7. Cómo insertar tablas

Las tablas son un elemento también muy habitual en una memoria de TFG, especialmente si de tipo experimental y, en consecuencia, ha de informar sobre resultados obtenidos de los experimentos realizados. Las tablas probablemente sean el tipo de elemento más complejo de  $\text{\LaTeX}$  por su flexibilidad, ya que pueden tener distribuciones no regulares y contener cualquier elemento en cada celdilla, incluso otras tablas.

### 1.7.1. Aspectos importantes al usar tablas

Las normas a tener presentes a la hora de introducir tablas en la memoria son las indicadas a continuación:

- Todas las tablas han de estar numeradas, usando una numeración consecutiva para toda la memoria o bien relativa a cada capítulo.
- Todas las tablas han de contar con un pie de tabla que describa claramente qué información facilita. No ocurre nada si el pie de tabla es extenso y ocupa varias líneas. Es especialmente importante describir qué hay en cada columna en caso de que los títulos de la primera fila no sean autoexplicativos<sup>4</sup>
- Todas las tablas han de estar referenciadas desde el texto de la memoria, al igual que las figuras.

---

<sup>4</sup>En ocasiones conviene poner un título de columna corto a las columnas, para conseguir que el ancho de la tabla no exceda del ancho de la página, usándose el pie para facilitar una descripción más pormenorizada.

- El formato de todas las tablas introducidas en la memoria será homogéneo, constando de:

1. Una línea horizontal que da inicio a la tabla
2. Una línea con los encabezados de cada columna en negrita
3. Una línea horizontal de separación entre encabezados y contenido
4. Tantas líneas como filas deba contener la tabla
5. Una línea horizontal de cierre de la tabla
6. El pie de tabla

Al igual que las figuras o los títulos de sección y subsección, las tablas permiten *romper* una sucesión de múltiples párrafos de texto haciendo más fácil la lectura de la memoria.

### 1.7.2. Cómo almacenar las tablas

Las tablas pueden introducirse directamente en el código fuente  $\text{\LaTeX}$ , pero en ocasiones puede interesar tenerlas almacenadas en archivos independientes, por ejemplo si han sido generadas por otro programa. En ese caso se guardarán en un archivo con extensión `.tex` y se incluirán en el capítulo que corresponda usando el comando `\input{ruta/archivo.tex}`.

### 1.7.3. Cómo insertar las tablas en el documento

El ámbito para incluir una tabla en la memoria es `tabular{}`<sup>5</sup>. Dentro de las llaves se indicará el número de columnas con que contará la tabla y su alineación, usando para ello los siguientes indicadores:

- `l`: columna de longitud variable con alineación a la izquierda
- `r`: columna de longitud variable con alineación a la derecha
- `c`: columna de longitud variable con alineación al centro
- `p{N.Mcm}`: columna de longitud fija  $N.M$  centímetros

El ancho de la tabla será la suma de los anchos de sus columnas. Debe tenerse en cuenta que ese ancho se extenderá todo lo necesario para acoger su contenido salvo para columnas de tipo `p`.

Dentro del ámbito `tabular` incluiremos los siguientes elementos por este orden:

1. `\toprule`: una línea horizontal que delimita el inicio de la tabla

---

<sup>5</sup>Recuerda que al ser un ámbito deberá ir delimitado con los comandos `\begin` y `\end`.

2. **Encabezado:** una fila de encabezado con los títulos de las columnas en negrita. Cada columna se separará de la siguiente con el carácter & y el final de la fila se marcará con \\
3. `\midrule`: una línea horizontal que separa el encabezado del contenido de la tabla
4. **Contenido:** tantas filas de contenido como sean precisas, separando el dato de cada columna del de la siguiente con el carácter & y marcando el final de cada fila con \\
5. `\bottomrule`: una línea horizontal para marcar el final de la tabla

Dado que la tabla debe contar además con un título y también un etiqueta, como las figuras, será preciso usar los comandos `\caption` y `\label` ya explicados. Todos ellos se unirán en un entorno `table`, tal y como muestra el ejemplo del Listado 1.3 que produce como resultado la Tabla 1.1.

```

1 \begin{table}[ht!]
2   \centering
3   \small
4   \def\arraystretch{1.5}
5   \begin{tabular}{llp{10cm}}
6     \toprule
7     \textbf{Componente} & \textbf{Requerimiento} & \textbf{Descripción} \\
8     \midrule
9     Procesador & Arquitect. x64 & Procesador con varios (\textit{cores}) para poder
10      ejecutar los distintos contenedores que forman el sistema \\
11
12     RAM & 64GB & Disponer de suficiente memoria RAM es importante para garantizar
13      un funcionamiento correcto del sistema \\
14
15     Disco & 10TB & Almacenamiento masivo suficiente para contener los datos. Se
16      recomienda una velocidad de al menos 7200rpm \\
17
18     GPU & CCC $\ge$ 3.5 & GPU para juegos con CUDA Compute Capability 3.5 o
19      posterior y al menos 8GB de memoria \\
20
21     Red & GbE & Gigabit Ethernet para el acceso del sistema a web \\
22   \bottomrule
23 \end{tabular}
24 \caption{Requisitos hardware del ordenador }
25 \label{Tabla.Requisitos}
26 \end{table}

```

Listado 1.3: Creación de una tabla

En este ejemplo se han usado, además de los ya descritos, algunos comandos  $\text{\LaTeX}$  adicionales: con `\centering` se consigue que la tabla, si no ocupa todo el ancho de la página, aparezca centrada; con `\small` se ajusta el tipo de letra de la tabla para que sea más pequeño que el texto general y, de esta forma, pueda ajustarse al espacio disponible; por último con `\def\arraystretch{1.5}` se modifica la separación por defecto de las líneas de la tabla, que es 1.0, incrementándola a 1.5.

Componente	Especificación	Descripción
Procesador	Arquitect. x64	Procesador de última generación con múltiples núcleos ( <i>cores</i> ) de ejecución para poder ejecutar los distintos contenedores que forman el sistema
RAM	64GB	Disponer de suficiente memoria RAM es importante para garantizar un funcionamiento correcto del sistema
Disco	10TB	Almacenamiento masivo suficiente para contener las imágenes, vídeos, modelos de IA y resto de elementos del sistema. Se recomienda una velocidad de al menos 7200rpm
GPU	CCC $\geq 3.5$	GPU para la ejecución de los modelos de IA con núcleos CUDA Compute Capability versión 3.5 o posterior y al menos 8GB de memoria
Red	GbE	Conexión Gigabit Ethernet para el acceso del sistema a fuentes de datos externas y la compartición de contenidos

Tabla. 1.1: Requisitos hardware del ordenador en el que se instalará el sistema

Es habitual que las tablas y figuras, cuando su tamaño es mayor que el espacio que resta en la página, pasen automáticamente al inicio de la página siguiente.  $\text{\LaTeX}$  siempre tratará de ajustar los contenidos para conseguir el mejor resultado posible, de forma que el texto quede ajustado llenando las páginas. Este párrafo de texto, por ejemplo, está dispuesto tras la Tabla 1.1, pero en el documento aparece dividido rodeando a la tabla, de forma que la página anterior no quede con un espacio vacío en la parte inferior. Puedes probar a mover párrafos de texto, poniéndolos delante o detrás del punto en el que se introduce la tabla/figura, para conseguir la disposición que te interese.

## 1.8. Cómo insertar fórmulas/ecuaciones

$\text{\LaTeX}$  se caracteriza por su versatilidad a la hora de tratar con símbolos, fórmulas y ecuaciones matemáticas, consiguiendo unos resultados de calidad superior. Esencialmente nos encontraremos con dos escenarios de uso distintos:

- La introducción de expresiones matemáticas en línea con el texto existente en un párrafo. En este caso se usarán el delimitador  $\$$  al inicio y final de la expresión, usando en su interior los comandos específicos de  $\text{\LaTeX}$  para esta tarea. Por ejemplo, con la expresión  $\$f(n) = n^5 + 4n^2 + 2\$$  conseguimos el resultado  $f(n) = n^5 + 4n^2 + 2$ .
- La introducción de ecuaciones de manera independiente, en cuyo caso deben ir numeradas y se les asignará una etiqueta para poder hacer referencia a ellas desde el texto. Con este fin se usará el ámbito `equation`. La numeración se generará automáticamente y aparecerá en el margen derecho, entre paréntesis.

El código mostrado en el Listado 1.4 genera como resultado la Ecuación 1.1. Observa el uso del comando `\label` para asignar una etiqueta que se ha usado en este párrafo de texto para hacer referencia a ella con el comando `\ref`.

```
1 \begin{equation}
2   P\left(A=2\middle|\frac{A^2}{B}>4\right)
3   \label{Eq.Prob1}
4 \end{equation}
```

Listado 1.4: Definición de una ecuación

$$P\left(A=2\middle|\frac{A^2}{B}>4\right) \quad (1.1)$$

Sobre el lenguaje que permite generar las expresiones matemáticas se podría escribir un libro completo. En WikiBooks ([en.wikibooks.org/wiki/LaTeX/Mathematics](http://en.wikibooks.org/wiki/LaTeX/Mathematics)) puedes encontrar una introducción que te solventará la mayoría de dudas al respecto.

## 1.9. Cómo insertar algoritmos

En una memoria de TFG es habitual tener que describir cómo funcionan los algoritmos usados o propuestos. Con ese fin lo más recomendable es facilitar un pseudo-código que describa dichos algoritmos. Es deseable que todos los algoritmos se describan con una sintaxis y formato similares, de forma que sean consistentes a lo largo de toda la memoria.

Por esa razón se recomienda usar el entorno `algorithm` a la hora de describir algoritmos. En este entorno usaremos los comandos `\caption` y `\label`, como haríamos con una figura o una tabla, y aparte un conjunto de comandos específicos para la definición de condicionales, bucles, etc., tal y como se aprecia en el Algoritmo 1. Este ha sido generado con el código  $\text{\LaTeX}$  que aparece en el Listado 1.5.

---

**Algoritmo 1:** Título del algoritmo descrito

---

**Result:** Resultado que devuelve algoritmo

---

inicialización;

**while** *condición* **do**

    instrucciones;

**if** *condición* **then**

        instrucciones1;

        instrucciones2;

**else**

        instrucciones3;

**end**

**end**

---

Observa que el título del algoritmo aparece en la parte superior, en lugar de en la inferior como ocurre con imágenes, tablas y otros elementos. Al igual que en las tablas, se usan líneas horizontales que delimitan las distintas partes del algoritmo.

```

1 \begin{algorithm}[H]
2 \SetAlgoLined
3 \KwResult{Resultado que devuelve algoritmo}
4   inicializacion\;
5   \While{condicion}{
6     instrucciones\;
7     \eIf{condicion}{
8       instrucciones1\;
9       instrucciones2\;
10    }{
11      instrucciones3\;
12    }
13  }
14 \caption{Titulo del algoritmo descrito}
15 \end{algorithm}

```

Listado 1.5: Descripción de un algoritmo

En cuanto a los comandos que pueden emplearse dentro del entorno `algorithm`, encontrarás información de referencia y abundancia de ejemplos en el manual del paquete `LATEX algorithm2e` ([osl.ugr.es/CTAN/macros/latex/contrib/algorithm2e/doc/algorithm2e.pdf](http://osl.ugr.es/CTAN/macros/latex/contrib/algorithm2e/doc/algorithm2e.pdf)), encargado de aportar el citado entorno y sus comandos específicos. En la mayoría de los casos solo precisarás usar un pequeño subconjunto de esos comandos para describir el algoritmo.

## 1.10. Cómo insertar código

Además de algoritmos descritos desde una perspectiva abstracta, mediante pseudo-código, también es probable que en la memoria sea preciso introducir listados de código. En las secciones previas aparecen múltiples listados, todos ellos han sido generados con el entorno `lstlisting`. Este toma, entre paréntesis, tres parámetros:

- `caption`: establece el título que aparecerá bajo el listado
- `language`: indica el lenguaje usado en el listado, de forma que se diferencien con colores y tipos de letra las distintas partes
- `label`: etiqueta a asignar para referenciar el listado

En el Listado 1.6 se muestra el fuente `LATEX` usado para producir el Listado 1.7. En el primero faltaría cerrar el entorno con el comando `\end{lstlisting}`.

```

1 \begin{lstlisting}[caption={Crear una tabla},language=SQL,label=List.CreaTabla]
2 CREATE TABLE Entradas(

```

```

3 id INTEGER PRIMARY KEY,
4 asiento INTEGER,
5 cliente CHAR(50)
6 );

```

Listado 1.6: Fuente LaTeX que genera el listado en lenguaje SQL

```

1 CREATE TABLE Entradas(
2 id INTEGER PRIMARY KEY,
3 asiento INTEGER,
4 cliente CHAR(50)
5 );

```

Listado 1.7: Creación de una tabla con SQL

El paquete que facilita el entorno `lstlisting` contempla el uso de decenas de lenguajes. La lista completa la tenemos en [en.wikibooks.org/wiki/LaTeX/Source\\_Code\\_Listings](http://en.wikibooks.org/wiki/LaTeX/Source_Code_Listings), de donde tomaríamos el nombre que es necesario asignar al parámetro `language` del entorno. En esa misma página también se describe cómo definir una configuración a medida para lenguajes que no estén soportados inicialmente. El formato por defecto para esta plantilla está definido en el archivo `Portada.tex`, junto con muchas otras definiciones.

## 1.11. Cómo citar textos y otros materiales

Durante el desarrollo del TFG es habitual recurrir al estudio de diversos materiales, entre los que se incluyen libros, artículos científicos y, por supuesto, páginas web con tutoriales e instrucciones. También durante la redacción de la memoria es posible que se usen definiciones tomadas literalmente o incluso diagramas y figuras.

En todos esos casos es imprescindible citar de manera adecuada la fuente de la que procede la información en que basamos nuestro trabajo o que reproducimos literalmente en la memoria.

### 1.11.1. Citas bibliográficas

Cualquier libro, artículo, contribución a congreso o recurso web que hayamos consultado para poder adquirir los conocimientos necesarios para desarrollar el TFG, ya sean teóricos sobre técnicas o prácticos sobre herramientas, deberíamos citarlos en nuestra memoria. La introducción de estas citas es un aspecto que se descuida en muchas ocasiones y que suele afectar negativamente a la evaluación de una memoria.

La introducción de citas bibliográficas en la memoria se lleva a cabo mediante el comando `\cite{}`, introduciendo en las llaves el identificador que se haya asignado a la referencia bibliográfica en el archivo `bibliografía.bib`. El proceso consta de varios pasos:

1. Obtener los datos bibliográficos en formato BibTeX:

- En Google Scholar ([scholar.google.es](https://scholar.google.es)) hacer clic en el icono en forma de comillas que aparece debajo del título del libro/artículo, a continuación elegir la opción **BibTeX**, seleccionar todo y copiar al portapapeles.
  - La revista/editorial en que se haya publicado el trabajo suele ofrecer la información bibliográfica en formato BibTeX.
  - Crear manualmente la entrada BibTeX con los datos bibliográficos que hayamos obtenido de la fuente de la que procede el material.
2. Introducir los datos bibliográficos en el archivo `bibliografía.bib`, asignando en la primera línea un identificador que nos sea fácil recordar por el tema y/o autor del material.
  3. Insertar en la ubicación que proceda de nuestra memoria la referencia bibliográfica, usando para ello el comando `\cite{}` y el identificador que se haya asignado en el paso previo.

Las entradas BibTeX almacenadas en el archivo `bibliografía.bib` tendrán un formato u otro dependiendo de que el material a citar sea un artículo científico, un libro, una contribución a un congreso o un recurso web. Cada entrada BibTeX se compone de múltiples atributos con datos:

- Por regla general, toda entrada deberá contar al menos con los atributos `title`, `author` y `year`, con los que se facilita el título del trabajo, su autor y año de publicación.
- En el caso de los libros también se debe incluir la editorial e ISBN, usando para ello los atributos `publisher` e `isbn`, respectivamente.
- Para los artículos indicar el volumen, número y páginas que ocupa en la revista, así como el nombre de esta, usando los atributos `volume`, `number`, `pages` y `journal`, respectivamente.
- En el caso de los recursos web es indispensable facilitar en el campo `url` la dirección web, así como indicar en el atributo `note` la fecha en que se verificó la disponibilidad del recurso.
- Siempre que sea posible, usar el atributo `doi` para facilitar el DOI (*Digital Object Identifier*) del documento.

Debemos tener presente que algunos servicios, como el mencionado Google Scholar, no facilita información completa en las entradas BibTeX que generan automáticamente. Por ello siempre debemos revisarlas y, en caso necesario, completar los datos que falten, buscándolos en el editor del libro, página web de la revista, etc.



A modo de ejemplo, en el Listado 1.8 se facilitan cuatro referencias bibliográficas, cada una de ellas correspondiente a uno de los tipos de entradas BibTeX mencionados. Estos ejemplos también los encontrarás en el archivo `bibliografía.bib`, por lo que puedes usarlos como plantilla para crear tus propias entradas en caso necesario.

```

1 @article{UnArticulo,
2   title={A Comprehensive and Didactic Review on Multilabel Learning Software Tools},
3   author={Charte, Francisco},
4   journal={IEEE Access},
5   volume={8},
6   pages={50330--50354},
7   year={2020},
8   publisher={IEEE},
9   doi={10.1109/ACCESS.2020.2979787}
10 }
11
12 @book{UnLibro,
13   title={Multilabel Classification: Problem Analysis, Metrics and Techniques},
14   author={Herrera, Francisco and Charte, Francisco and Rivera, Antonio J and del Jesus, Mar
15     {\'\i}a J},
16   year={2016},
17   publisher={Springer},
18   ISBN={978-3-319-41110-1},
19   doi={10.1007/978-3-319-41111-8}
20 }
21
22 @inproceedings{UnCongreso,
23   title={A first approach to deal with imbalance in multi-label datasets},
24   author={Charte, Francisco and Rivera, Antonio and del Jesus, Mar{\'\i}a Jos{\'e} and
25     Herrera, Francisco},
26   booktitle={International Conference on Hybrid Artificial Intelligence Systems},
27   pages={150--160},
28   year={2013},
29   organization={Springer},
30   doi={10.1007/978-3-642-40846-5\_16}
31 }
32
33 @online{UnaWeb,
34   author = {Charte, Francisco},
35   title = {Cómo analizar la distribución de los datos con R},
36   year = 2019,
37   url = {https://fcharte.com/tutoriales/20170114-DistribucionDatosR/},
38   note = {comprobado en 2020-09-30}
39 }

```

Listado 1.8: Entradas BibTeX de distintos tipos

Teniendo las entradas BibTeX ya preparadas, solo quedaría introducir las referencias en los puntos adecuados del texto, tal y como se explicó antes. Un párrafo como el mostrado en el Listado 1.9 generaría el resultado que se aprecia justo a continuación. Además, introducirá la información bibliográfica completa en la lista de referencias, al final de la memoria, con un formato adecuado.

```

1 Como se explica en~\cite{UnArticulo} y se amplía en el libro~\cite{UnLibro}, las técnicas
2 que se presentaron en~\cite{UnCongreso} pueden ponerse en práctica siguiendo el

```

## Listado 1.9: Párrafo en el que se citan varios trabajos

Como se explica en [Charte \(2020\)](#) y se amplía en el libro [Herrera et al. \(2016\)](#), las técnicas que se presentaron en [Charte et al. \(2013\)](#) pueden ponerse en práctica siguiendo el tutorial [Charte \(2019\)](#)

### 1.11.2. Citas textuales

Si además de emplear un material como base para el estudio y desarrollo de nuestro TFG, caso en el que lo citaremos según lo que acaba de explicarse, también hemos tomado alguna parte del mismo para introducirlo literalmente en la memoria, hemos de tener en cuenta las siguientes normas:

- Únicamente pueden citarse textualmente pequeñas porciones de un trabajo, habitualmente no más de un párrafo y en ningún caso páginas completas.
- El texto citado ha de aparecer claramente diferenciado en el texto de la memoria, para lo cual usaremos habitualmente el entorno `\displayquote` y pondremos el texto en cursiva, consiguiendo un resultado como el siguiente:

*Learning, like intelligence, covers such a broad range of processes that it is difficult to define precisely. A dictionary definition includes phrases such as “to gain knowledge, or understanding of, or skill in, by study, instruction, or experience,” and “modification of a behavioral tendency by experience.” Zoologists and psychologists study learning in animals and humans. In this book we focus on learning in machines.*

- La cita textual ha de ir acompañada de la correspondiente cita bibliográfica, de forma que sea posible determinar el origen del material de manera inequívoca.
- En general no es recomendable incluir en la memoria ninguna ilustración, diagrama o figura cuyos derechos de uso no estén claros. Si la licencia de uso es *Creative Commons* se puede incluir la figura indicando tanto la licencia como la procedencia. En cualquier otro caso es necesario obtener permiso del propietario para la reproducción del material. La alternativa es construir nuestra propia ilustración o diagrama indicando que se ha tomado como base una fuente existente y facilitando la correspondiente referencia.

Ten siempre presente que tu memoria de TFG va a publicarse y cualquiera tendrá acceso al trabajo que has hecho durante mucho tiempo. También es habitual que la memoria se sometida a herramientas de detección de plagio. Por todo ello, es importante ser honesto y no usar nunca trabajo de terceros sin citarlo apropiadamente.

## 1.12. Cómo introducir ciertos caracteres en L<sup>A</sup>T<sub>E</sub>X

Al trabajar con fuentes L<sup>A</sup>T<sub>E</sub>X empleamos un conjunto de caracteres que tienen un significado especial, como es el carácter `\` para con el que se inician los comandos, el carácter `$` que delimita las expresiones matemáticas, el carácter `%` para introducir comentarios, los símbolos `_` y `^` para escribir subíndices y superíndices en fórmulas matemáticas, etc. Incluso algunos símbolos, como es el caso de las comillas dobles", pueden tener un resultado no deseado<sup>6</sup> en el documento.

En general, usaremos el símbolo `\` como carácter de escape, para que el carácter que dispongamos a continuación aparezca como tal en el texto. No obstante hay excepciones: para incluir una barra invertida no podemos usar `\\`, porque eso indica un salto de línea.

Si tienes problemas para introducir algún carácter puedes recurrir a [en.wikibooks.org/wiki/LaTeX/Special\\_Characters](https://en.wikibooks.org/wiki/LaTeX/Special_Characters) para saber cómo debes introducirlo en tu memoria.

## 1.13. Cómo obtener esta plantilla

Esta plantilla está disponible públicamente en el repositorio Github [TFGLatexTemplate](#) de su autor, Francisco Charde Ojeda.

## 1.14. Cómo configurar la plantilla con nuestros datos

A fin de generar nuestra memoria de TFG usando esta plantilla, aprovechando toda la configuración de formatos ya adaptada a la normativa de la EPSJ, tendremos que comenzar abriendo el archivo `main.tex` para configurar nuestros datos personales. Para ello localiza el bloque L<sup>A</sup>T<sub>E</sub>X que aparece en el Listado 1.10 y sigue las instrucciones indicadas en los comentarios.

```
1 % ==== Introducir aquí el nombre del estudiante
2 \def\Estudiente{Nombre1 Nombre2 Apellido1 Apellido2}
3
4 % ==== Introducir aquí el nombre de los tutores. Si solo hay uno dejar las llaves de \TutorB vacías
5 \def\TutorA{Nombre y apellidos del tutor 1}
6 \def\TutorB{Nombre y apellidos del tutor 2}
7
8 % ==== Introducir aquí el título de completo y abreviado (para las cabeceras) del TFG
9 \def\TituloTFG{El título del TFG tal y como aparece en la propuesta original}
10 \def\TituloAbreviado{Título abreviado para el encabezado}
11
12 % ==== Introducir aquí el mes y año de presentación del TFG
13 \def\Fecha{junio de 2021}
```

Listado 1.10: Variables a establecer en `main.tex`

<sup>6</sup>En el fuente L<sup>A</sup>T<sub>E</sub>X se ha usado el carácter `"` para iniciar el entrecomillado, pero esto ha tenido como efecto convertir la `"c"` inicial en una ç.

Solo necesitas establecer el contenido de estas variables para que la portada, página de autorización y los encabezados muestren los datos correctos. Habitualmente esto es todo lo que necesitarás cambiar en el archivo `main.tex`. No obstante, si no quieres incluir al inicio de la memoria alguno de los índices que aparecen por defecto, como los de algoritmos, listados, etc., también deberás eliminar o comentar (colocar delante el carácter `%`) las líneas correspondientes.

## 1.15. En cuanto al contenido de los capítulos

Una vez que hayas establecido la configuración con tus datos, el paso siguiente será escribir el contenido de la memoria. Para ellos se definirán y completarán los capítulos que correspondan, según las recomendaciones dadas en los dos siguientes apartados. Como toda recomendación, dicha estructura podrá adecuarse a los aspectos específicos de cada TFG.

Cada capítulo de la memoria se almacenará en un archivo independiente, con extensión `.tex`, que se insertará desde `main.tex` usando el comando `\input`. En la versión original de dicho archivo, tal y como se facilita en esta plantilla, ya se incluyen los capítulos del primer tipo de TFG. El estudiante deberá agregar o eliminar los que precise, por ejemplo si su TFG es de tipo *Proyecto de ingeniería* en lugar de *Trabajo teórico/experimental*.

### 1.15.1. TFG de tipo *Trabajo teórico/experimental*

Una memoria de TFG de este tipo se estructura en los capítulos indicados a continuación. Dado que esta plantilla se ajusta a ese tipo concreto, además del nombre de los capítulos también se señala el nombre del archivo `.tex` en que se encuentra.

#### 1. **Introducción** - `Introduccion.tex`<sup>7</sup>

La introducción al TFG, como su propio nombre denota, ha de servir como un acercamiento general y desde una perspectiva de alto nivel al trabajo realizado. Debe explicarse la motivación que nos ha llevado a abordar el problema en cuestión, su importancia, cómo se ha tratado hasta el momento, qué creemos que podemos aportar, etc., todo ello sin entrar demasiado en detalle. Es habitual indicar en la parte final de la introducción la estructura de la memoria, enumerando sus capítulos con una breve descripción de lo que se explica en cada uno de ellos. También puedes incluir un índice o tabla, al final de este capítulo, conteniendo todos los acrónimos y términos que uses en la memoria, de forma que sea fácil para quien la lea saber qué significan<sup>8</sup>. También se puede facilitar dicha lista en una apéndice al final de la memoria.

---

<sup>7</sup>En la plantilla este archivo contiene las instrucciones de cómo usar  $\text{\LaTeX}$  y la propia plantilla, por lo que deberás eliminar dicho contenido o bien crear tu propio archivo `Introduccion.tex` desde cero.

<sup>8</sup>La alternativa es definirlos la primera vez que aparezcan en el texto, por ejemplo: «En este TFG (*Trabajo Fin de Grado*) se aborda ...».

## 2. **Antecedentes** - Antecedentes.tex

Este capítulo tiene como objetivo demostrar que has estudiado el tema que se aborda en el TFG, describiendo primero el problema con todos los detalles necesarios y después las técnicas que se han empleado hasta el momento para tratarlo y las que propones usar tú. Un aspecto fundamental de este capítulo es la bibliografía: has de recopilar los artículos científicos más relevantes sobre el tema en cuestión, consultar libros sobre las técnicas a usar, etc., referenciando todo ello de manera adecuada.

## 3. **Objetivos** - Objetivos.tex

El título del capítulo lo dice todo: ¿cuáles son los objetivos que se persiguen al desarrollar este TFG? Habitualmente se expondrá un objetivo general, desde una perspectiva de alto nivel, y después se descompondrá en tantos objetivos específicos como sea preciso, detallando cada uno de ellos al máximo. En teoría, solo leyendo este capítulo debería obtenerse una idea muy clara de qué va a hacerse (se ha hecho) en el TFG.

## 4. **Materiales y métodos** - MaterialMetodos.tex

Para alcanzar los objetivos establecidos en el capítulo previo será preciso usar unos materiales: por ejemplo los datos que servirán para generar los modelos, y también unos métodos: las técnicas y herramientas que servirán para obtener los resultados. Todos esos aspectos han de quedar reflejados en este capítulo hasta el más mínimo detalle, explicando, por ejemplo, cómo se han obtenido los datos, si ha sido necesario prepararlos de alguna manera antes de poder usarlos; qué técnicas/algoritmos van a emplearse y por qué, cómo va a realizarse la ejecución (configuración hardware y software), etc.

## 5. **Resultados** - Resultados.tex

Como salida de aplicar los métodos a los materiales se obtienen resultados. La finalidad de este capítulo doble: por una parte se deben reflejar todos los resultados obtenidos, usualmente en tablas en gráficas, por otra, ha de efectuarse un análisis pormenorizado de dichos resultados a fin de determinar su valor. Siempre que sea posible, deberían realizarse comparaciones de resultados entre distintos métodos, ya los hayamos aplicado nosotros o hayan sido previamente publicados en la bibliografía citada en los antecedentes.

## 6. **Conclusiones** - Conclusiones.tex

El último capítulo de la memoria ha de presentar una recopilación de todo el trabajo realizado, los resultados obtenidos y un análisis que determine, y deje claro, si se han alcanzado las metas que se establecieron en la propuesta inicial del TFG que aprobó la correspondiente comisión. Además, también es habitual incluir una sección

describiendo potenciales mejoras que podrían efectuarse o trabajos futuros que podrían derivarse de lo hecho en el TFG.

### 1.15.2. TFG de tipo *Proyecto de ingeniería*

Una memoria de TFG de este tipo se estructura en los capítulos indicados a continuación. Junto al título de cada capítulo se sugiere el nombre que podría tener el archivo correspondiente.

#### 1. **Introducción** - `Introduccion.tex`

El contenido de este primer capítulo será análogo al mencionado para el tipo de TFG anterior, por lo que la descripción ya facilitada es aplicable aquí.

#### 2. **Planificación y presupuesto** - `Planificacion.tex`

En el segundo capítulo se ha de facilitar una planificación tanto temporal como económica para el desarrollo del proyecto, con una enumeración de todos los recursos precisos para alcanzar su consecución. Es habitual referenciar la normativa legal y técnica que se ha seguido para el desarrollo del proyecto, así como la metodología empleada para su organización y gestión. Ha de documentarse el proceso de cálculo para la estimación del tamaño y esfuerzo del software a construir, información a partir de la cual se derivará la planificación temporal —lo habitual es que se refleje en un diagrama de Gantt— y el presupuesto económico.

#### 3. **Análisis de requisitos** - `Requisitos.tex`

A continuación hay que documentar las entrevistas mantenidas con los hipotéticos usuarios/clientes, a partir de las cuales se identificarán los requisitos, tanto funcionales como no funcionales, del sistema. Estos se analizarán, se especificarán mediante los artefactos adecuados —diagrama de casos de uso, historias de usuario, modelo del dominio, etc.— y se validarán con los clientes. Es aconsejable seguir las prácticas recomendadas para especificación de requisitos en la [norma 29148-2018 de ISO/IEC/IEEE](#).

#### 4. **Diseño del sistema** - `Diseno.tex`

En el cuarto capítulo, a partir de los requisitos previamente documentados, se procederá a realizar el diseño detallado del sistema. Aquí hay que considerar tres partes bien diferenciadas: el núcleo del sistema, su interfaz de usuario (si es que cuenta con una) y los datos sobre los que opera. Para el **núcleo del sistema** se comienza con diagramas UML de alto nivel, como los de arquitectura y paquetes, para después construir el diagrama o diagramas de clases. Por cada operación del sistema se detallarán en un diagrama de secuencia los pasos de que consta y los componentes del sistema que

intervienen. También pueden emplearse diagramas de actividad u otros tipos que se consideren apropiados. La documentación de la **interfaz de usuario** se realizará con *wireframes* o *mockups*, con el detalle necesario para facilitar su posterior implementación. En cuanto a **los datos** sobre los que operará el software, se detallará su estructura con diagramas entidad/relación o equivalentes, según el tipo de los propios datos. Este capítulo servirá también para mencionar los patrones, arquitectónicos (p.e. MVC) y de diseño (p.e. *singleton*), que se empleen en el proyecto.

## 5. Implementación - `Implementacion.tex`

Los capítulos previos ofrecen una visión del software a desarrollar en la que no se hace asunción alguna respecto a los lenguajes de programación, bibliotecas de servicios y otras herramientas que servirían para realizar la implementación. Será en este capítulo donde se enumeren las potenciales alternativas, se evalúen sus ventajas y desventajas y, finalmente, se haga una elección justificada de las que se utilizarán. Asimismo, se detallarán las iteraciones del proceso de implementación, estructura del proyecto software, las guías de estilo que se han seguido para la codificación y documentación del código, archivos de configuración si existen (formato y parámetros), proceso de compilación (*Makefile*), etc.

## 6. Pruebas - `Pruebas.tex`

Las etapas por las que transcurra el desarrollo del software deberían ir acompañadas de sus respectivas baterías de pruebas. Es habitual incluir pruebas unitarias (*unit testing*), de integración y de regresión. Este capítulo servirá para documentar tanto las herramientas empleadas para llevar a cabo esas pruebas como los test en sí mismos.

## 7. Despliegue y mantenimiento - `Despliegue.tex`

En caso de que sea preciso, se incluirá este capítulo para describir los pasos a seguir para el despliegue y puesta en explotación del software. Es especialmente importante si se depende de alojamiento (*hosting*) externo o cualquier servicio de terceros. También es importante, si procede, enumerar los aspectos a tener presentes de cara a mantener la aplicación tras su puesta en explotación, detallando las operaciones que esta necesite: actualización de parámetros, iniciación periódica, etc.

## 8. Manual de instalación y uso - `Manuales.tex`

La última parte de la memoria del TFG la componen los manuales de instalación del software y de uso del mismo. A diferencia de los anteriores, este material ha de dirigirse al usuario final de la aplicación y no a personas con experiencia en desarrollo o programación, adecuando el vocabulario y nivel para facilitar que cualquiera pueda llevar a cabo la instalación y proceder al uso del software.

En el caso de los TFG de este tipo, varios de los capítulos deben aportar diagramas de diferentes niveles: modelo del dominio, arquitectura del sistema, diagrama de paquetes, de clases, secuencia, E/R, etc. Se recomienda usar la [herramienta PlantUML](#) con este fin, un programa que se integra cómodamente en editores como VSCode y que permite definir los diagramas mediante archivos de código, lo cual hace posible versionarlos —es aconsejable usar `git` durante todo el proceso— y generar la imagen del diagrama en diferentes formatos<sup>9</sup>.

### 1.15.3. Otros archivos del proyecto

Además de los archivos anteriores, correspondientes a los capítulos, también debes editar el contenido del archivo `Agradecimientos.tex` para escribir tus propios agradecimientos y dedicatoria.

## 1.16. Consejos sobre redacción

Para terminar estas instrucciones, y aunque no tiene que ver con la plantilla  $\text{\LaTeX}$  ni aspectos relativos a la normativa de realización de TFG de la EPSJ, quiero incidir en un aspecto de vital importancia: la calidad de la redacción.

Tanto para el tribunal que evaluará tu TFG como para las personas que puedan leerlo en el futuro, la corrección del texto será siempre un reflejo de la calidad del trabajo que has llevado a cabo, aunque esta no sea la realidad. Es habitual que un muy buen trabajo vea su evaluación penalizada a causa de una pobre redacción de la memoria. Por el contrario, un trabajo menos brillante pero con una presentación perfecta en la memoria puede ser mejor valorado.

Sin ánimo de ser exhaustivo, porque tampoco es la finalidad de estas instrucciones, he aquí una lista de aspectos a tener presentes en la redacción:

- **Ortografía:** posiblemente no haya nada que cause una peor impresión al leer una memoria que encontrar errores de ortografía. Extrema el cuidado en este sentido: revisa las normas de uso de *b* y *v*, de *g* y *j*, de *m* y *n*, de uso de la *h*, las relativas a las tildes, etc. Tan malo es el defecto como el exceso, por ejemplo poniendo tilde en términos como los pronombres (*este*, *esta*, *estas*) que la normativa indica desde hace años que no deben llevar tilde. Ante la duda usa el Diccionario de la RAE ([buscon.rae.es](http://buscon.rae.es)) y las consultas de la Fundéu ([www.fundeu.es/consultas](http://www.fundeu.es/consultas)).
- **Gramática:** usar los tiempos verbales correctos, respetar las concordancias de género y número y otros aspectos gramaticales es también vital para facilitar la correcta

---

<sup>9</sup>Es aconsejable que todas los diagramas e imágenes a incluir en el documento se guarden en formato vectorial, por ejemplo PDF, EPS o SVG, en lugar de como mapa de bits, a fin de garantizar una calidad óptima de los mismos en la memoria.



comprensión de las explicaciones dadas en la memoria. Una de estas normas nos dice que a los acrónimos y abreviaciones no se les añade la *s* final para formar el plural, sino que el número lo denota el determinante o el contexto, por ejemplo «En los TFG de este curso ...», en lugar de «En los TFGs de este curso ...». Otra regla es la de no comenzar con mayúsculas tras dos puntos, salvo para nombres propios, tal y como ves en esta misma sección, en la que tras el título en negrita y los dos puntos se continúa usando minúsculas.

- **Vocabulario:** el vocabulario usado al redactar denota diversas capacidades por parte del autor de la memoria. En primer lugar, el uso de términos especializados, específicos y exactos, del campo en el que se está trabajando indica que se conoce la materia, se ha leído sobre el tema. En segundo, el uso de un vocabulario variado, evitando repetir las mismas palabras y términos en un corto espacio, indica riqueza en el uso del lenguaje. Por último, debe tenerse presente que la memoria de un TFG es un documento formal y que, en consecuencia, debe evitarse el uso de términos demasiado coloquiales. Por ejemplo, usar «Según se ha descrito en ...» en lugar de «Como se ha comentado en ...».
- **Estructuración:** la memoria al completo, pero también cada uno de sus capítulos de manera aislada y cada sección de cada capítulo, deben redactarse como una relación de ideas interconectadas entre sí y que tienen sentido. Nada hay más desconcertante que leer una memoria en la que cada párrafo trata sobre ideas inconexas con lo que tiene a su alrededor, en párrafos anteriores y posteriores, provocando una sensación de que va saltándose de un tema a otro sin orden alguno. Por ello es importante llevar a cabo una planificación previa del contenido de cada capítulo y de cada sección, de forma que, al igual que durante el desarrollo de un software, se maximice la cohesión y se minimice el acoplamiento.

Aunque es algo totalmente fuera del ámbito de estas instrucciones, una última recomendación: la mejor forma de aprender a redactar correctamente estriba en leer mucho, especialmente literatura.



## Capítulo 2

# ANTECEDENTES

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac

pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

## Capítulo 3

### OBJETIVOS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac

pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

## Capítulo 4

# MATERIALES Y MÉTODOS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac

pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



## Capítulo 5

# RESULTADOS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac

pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

## Capítulo 6

# CONCLUSIONES

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac

pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

# Bibliografía

Charte, F. (2019). Cómo analizar la distribución de los datos con R. comprobado en 2020-09-30.

Charte, F. (2020). A comprehensive and didactic review on multilabel learning software tools. *IEEE Access*, 8:50330–50354.

Charte, F., Rivera, A., del Jesus, M. J., and Herrera, F. (2013). A first approach to deal with imbalance in multi-label datasets. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 150–160. Springer.

Herrera, F., Charte, F., Rivera, A. J., and del Jesus, M. J. (2016). *Multilabel Classification: Problem Analysis, Metrics and Techniques*. Springer.

