



Escuela Técnica Superior
de Ingeniería Informática

Grado en Ingeniería de Computadores

Curso 2021-2022

Trabajo Fin de Grado

**COMPARATIVA ENTRE LAS API DE SPARK EN
SCALA Y PYTHON**

Autor: Oscar Nydza Nicpoñ

Tutor: Juan Manuel Serrano Hidalgo

Agradecimientos

Breves agradecimientos o dedicatoria.

Resumen

Breve resumen del Trabajo de Fin de Grado (TFG). Recomendable entre 250-300 palabras, conteniendo los principales objetivos y resultados derivados del mismo.

Palabras clave:

- Python
- Ciberseguridad
- Aprendizaje automático (pueden ser varias)
- ...

Índice de contenidos

| | |
|---|-----------|
| Índice de figuras | X |
| Índice de códigos | XII |
| 1. Introducción | 1 |
| 1.1. Contexto y alcance | 1 |
| 1.2. Estructura del documento | 1 |
| 1.2.1. Trabajos de grados en informática | 2 |
| 1.2.2. Trabajos del grado en matemáticas | 2 |
| 2. Objetivos | 3 |
| 2.1. Primera sección | 3 |
| 2.2. Segunda sección | 3 |
| 3. Descripción Informática | 5 |
| 3.1. Fuentes de datos | 5 |
| 3.2. Programación de queries en PySpark | 5 |
| 3.3. Programación de queries en Scala/Spark | 6 |
| 3.4. Despliegue en AWS EMR | 8 |
| 3.4.1. Código | 9 |
| 4. Experimentos / Validación | 10 |
| 4.1. Consultas Realizadas | 12 |
| 4.1.1. Descripción General | 12 |
| 4.2. Análisis de requisitos no funcionales | 15 |
| 5. Conclusiones y trabajos futuros | 16 |
| 5.1. Texto de relleno | 16 |
| Bibliografía | 22 |
| Apéndices | 24 |
| A. Este es el primer apéndice | 26 |

| | |
|-----------------------------------|----|
| A.1. Ejemplo de sección | 26 |
|-----------------------------------|----|

Índice de figuras

| | |
|--|---|
| 3.1. Logo de la Universidad. | 6 |
| 3.2. Ejemplo con varias figuras. Demostración visual del teorema de Pitágoras. En (a) tenemos un triángulo rectángulo con hipotenusa c y catetos a y b . En (b) se muestra tres copias escaladas del mismo triángulo. El verde se ha escalado por a , el rojo/rosa por b , y el azul por c . En (c) se juntan los triángulos de (b) para formar un rectángulo cuya base es c^2 , pero también $a^2 + b^2$. Por tanto, $a^2 + b^2 = c^2$ | 8 |

Índice de códigos

| | |
|---|---|
| 3.1. Título del algoritmo/código. | 9 |
|---|---|

1

Introducción

Se puede añadir texto antes de empezar la primera sección.

1.1. Contexto y alcance

Contexto. Situar al lector. Objetivo general y alcance del trabajo.

1.2. Estructura del documento

La estructura del TFG no es fija. El tutor indicará una estructura adecuada dependiendo del trabajo concreto.

Se puede incluir dentro de cada apartado secciones adicionales. La copia en papel de la memoria del TFG será encuadernada en pasta dura de color azul (p.e. encuadernación tipo chanel). La portada, que puede ser una pegatina transparente, seguirá el modelo que se adjunta, que incluye el escudo y nombre de la URJC, la titulación cursada por el alumno, el curso académico, el título del TFG, el autor y el o los directores/tutores.

1.2.1. Trabajos de grados en informática

Una posible estructura de la memoria final asociada con cada TFG podría ser la siguiente (leed la normativa de TFG):

1. Introducción
2. Objetivos (incluyendo descripción del problema, estudio de alternativas y metodología empleada)
3. Descripción informática (puede incluir especificación, diseño, implementación y pruebas).
4. Experimentos / validación
5. Conclusiones (incluyendo los logros principales alcanzados y posibles trabajos futuros)
6. Bibliografía
7. Apéndices

1.2.2. Trabajos del grado en matemáticas

Una posible estructura de la memoria final asociada con cada TFG podría ser la siguiente:

1. Introducción
2. Objetivos (incluyendo descripción del problema, estudio de alternativas y metodología empleada)
3. Material y métodos / Metodología / Cuerpo del trabajo (describir las metodologías empleadas en el desarrollo del TFG o el desarrollo del mismo en caso de ser un trabajo de recopilación bibliográfica sobre un tema).
4. Resultados (opcional, dependiendo del tipo de trabajo desarrollado)
5. Conclusiones (incluyendo los logros principales alcanzados y posibles trabajos futuros)
6. Bibliografía
7. Apéndices

2

Objetivos

Objetivos generales y específicos del trabajo.

2.1. Primera sección

En este capítulo se pueden añadir secciones, pero no son obligatorias en capítulos cortos.

2.2. Segunda sección

El objetivo de este documento es proporcionar una plantilla de \LaTeX para TFG. No debe usarse como sustituto de la normativa de TFG aprobada por la ETSII.

3

Descripción Informática

3.1. Fuentes de datos

Esto es una referencia bibliográfica [1]. Se recomienda leer “The Not So Short Introduction to L^AT_EX” [2] (existen versiones más modernas).

3.2. Programación de queries en PySpark

Gracias a la ecuación de Euler ($e^{\pm i\theta} = \cos \theta \pm i \sin \theta$) podemos ver la relación entre varias de las constantes matemáticas más importantes:

$$e^{i\pi} + 1 = 0.$$

Si una ecuación se va a referenciar es necesario numerarla:

$$\Phi(k) = \frac{2}{|R(k)|(|R(k)| - 1)} \sum_{i,j \in R(k)} a_{ij}. \quad (3.1)$$

Posteriormente se hace referencia a la ecuación a través de su etiqueta (label). Por ejemplo, la anterior ecuación (3.1).

Tabla 3.1: Título de la tabla.

| | Subs. | Students | A | PE | WA | RE | CTE | IF | TLE | All |
|---------|-------|----------|------|------|------|------|------|------|------|------|
| Ex. 1 | 104 | 44 | 1.27 | 0 | 0.55 | 0.23 | 0.20 | 0.11 | 0 | 2.36 |
| Ex. 2 | 118 | 37 | 0.92 | 0 | 0.92 | 0.27 | 0.49 | 0.59 | 0 | 3.19 |
| Ex. 3 | 100 | 28 | 1.21 | 0.39 | 1.18 | 0.54 | 0.14 | 0.07 | 0.04 | 3.57 |
| Ex. 4 | 78 | 25 | 1.08 | 0.84 | 0.52 | 0.40 | 0.24 | 0.04 | 0 | 3.12 |
| Ex. 5 | 116 | 31 | 1.48 | 0.10 | 0.77 | 0.32 | 0.42 | 0.19 | 0.45 | 3.74 |
| Ex. 6 | 213 | 32 | 1.06 | 0.34 | 3.81 | 0.56 | 0.69 | 0.06 | 0.13 | 6.66 |
| Ex. 7 | 116 | 34 | 1.35 | 0.38 | 0.38 | 0.68 | 0.62 | 0 | 0 | 3.41 |
| Average | 120.7 | 33 | 1.20 | 0.26 | 1.14 | 0.42 | 0.40 | 0.16 | 0.08 | 3.66 |



Figura 3.1: Logo de la Universidad.

Problema de optimización:

$$\begin{aligned}
 &\text{minimizar} && \mathbf{1}^\top \mathbf{t} \\
 &\mathbf{t} \in \mathbb{R}^n, \mathbf{p} \in \mathbb{R}^m && \\
 &\text{sujeto a} && -\mathbf{t} \preceq \mathbf{V}\mathbf{p} - \mathbf{x} \preceq \mathbf{t},
 \end{aligned} \tag{3.2}$$

3.3. Programación de queries en Scala/Spark

Las tablas y figuras deben presentarse en el texto, referenciadas y numeradas. La descripción de una figura debe ir posicionada debajo de la misma. Las descripciones de tablas pueden aparecer encima o debajo de las mismas (pero de forma consistente en todo el documento).

En las tablas se recomienda evitar líneas verticales y usar pocas horizontales.

La figura 3.1 se utiliza en la portada. \LaTeX ubica automáticamente las tablas y figuras. Para ello emplea reglas basadas en la experiencia de profesionales de la edición de textos. Podemos forzar su ubicación, pero en general es recomendable usar la ubicación sugerida por el sistema \LaTeX . Usad gráficos vectoriales siempre

Tabla 3.2: Tabla rotada. Factor groupings for the Mooshak questionnaire.

| Factor | Interpretation / Items* (loadings) | Median | Mode |
|------------------|--|--------|------|
| 1 | Students' perception of Mooshak towards its helpfulness in learning | | |
| (21.17 %) | m10. Mooshak has forced me to implement programs more carefully (0,849) | 4 | 4 |
| $\alpha = 0.922$ | m6. Mooshak has helped me improve as a programmer (0,819) | 3 | 4 |
| | m5. Mooshak has made me more aware of the need to write correct code (0,781) | 3 | 3 |
| | m1. Mooshak has forced me to program more responsibly (0,713) | 3 | 3 |
| | m15. The specifications regarding the exercises used with Mooshak are adequate (0,687) | 3 | 3 |
| | m18. Mooshak helps to measure my current programming skills (0,680) | 2.5 | 3 |
| 2 | Disposition towards using Mooshak | | |
| (17.93 %) | m24. I would be willing to participate in a programming contest using Mooshak, with similar exercises to the ones seen throughout the course (0,807) | 2 | 1 |
| $\alpha = 0.897$ | m13. Using Mooshak in the final exams is a good idea (0,748) | 2 | 1 |
| | m14. I would like to use Mooshak or a similar tool in the future (0,734) | 3 | 1 |
| | m17. Knowing Mooshak can motivate me to take part in a programming contest (0,655) | 2 | 1 |
| | m9. It would have been useful to use Mooshak from the first programming course (0,527) | 2.5 | 1 |
| | m16. Using Mooshak in the course has been interesting (0,522) | 3 | 4 |
| 3 | Effect of Mooshak's feedback in the tool's usefulness | | |
| (14.84 %) | m12. Mooshak's feedback is adequate (0,832) | 2 | 1 |
| $\alpha = 0.836$ | m3. Using Mooshak has increased my workload considerably (0,693) | 4 | 4 |
| | m7. If Mooshak does not accept my code I feel motivated to find and fix the errors (0,691) | 2 | 3 |
| | m8. In general, using Mooshak has been a good idea (0,666) | 3 | 4 |
| 4 | Mooshak's effect on persistence | | |
| (11.20 %) | m23. When Mooshak does not accept my code I get discouraged and I abandon the exercise (0,848) | 3 | 3 |
| $\alpha = 0.705$ | m22. Mooshak has been a waste of time (0,597) | 2 | 2 |
| | m25. Once a program has passed Mooshak's tests, I rewrite it in order to enhance it (0,559) | 2 | 2 |
| 5 | Students' perception of Mooshak's features | | |
| (10.87 %) | m20. Even if it is not related to the grade, I feel satisfied if I am one of the first students to complete an exercise (0,729) | 2 | 2 |
| $\alpha = 0.742$ | m19. I value the fact that a tool like Mooshak returns feedback in real time about the correction of my programs (0,650) | 3.5 | 4 |

*Measured on a 5-point Likert scale (1: strongly disagree; 2: disagree; 3: neutral; 4: agree; 5: strongly agree).

| | | | |
|------------------|-------------|----------------|------------------|
| numeric literals | integers | in decimal | 8743 |
| | | in octal | 0o7464 |
| | | | 00103 |
| | | in hexadecimal | 0x5A0FF |
| | 0xE0F2 | | |
| | fractionals | in decimal | 140.58 |
| | | | 8.04e7 |
| | | | 0.347E+12 |
| 5.47E-12 | | | |
| 47e22 | | | |
| char literals | | | 'H' |
| | | | '\n' |
| | | | '\x65' |
| string literals | | | "bom dia" |
| | | | "ouro preto\nmg" |

Tabla 3.3: Tabla con “multicolumnas” y “multifilas”.

(a)

(b)

(c)

Figura 3.2: Ejemplo con varias figuras. Demostración visual del teorema de Pitágoras. En (a) tenemos un triángulo rectángulo con hipotenusa c y catetos a y b . En (b) se muestra tres copias escaladas del mismo triángulo. El verde se ha escalado por a , el rojo/rosa por b , y el azul por c . En (c) se juntan los triángulos de (b) para formar un rectángulo cuya base es c^2 , pero también $a^2 + b^2$. Por tanto, $a^2 + b^2 = c^2$.

que podáis.

3.4. Despliegue en AWS EMR

Normalmente no tendremos que insertar saltos de página, salvo para forzar que los capítulos empiecen en páginas impares, con

`\blankpage`

En cualquier caso, podemos introducir un salto de página con el comando

`\newpage`

.

Código 3.1: Título del algoritmo/código.

```

1 def sum_list_limits_1 (a, lower, upper):
2     if lower > upper:
3         return 0
4     else:
5         return a[upper] + sum_list_limits_1 (a, lower, upper -
        1)

```

Algoritmo 1 *Additional Louvain* **input**=(A , \mathcal{M}) **output**= P

```

1:  $\forall i \in V$ , let  $i$  be an isolated community
2:  $o = \text{permutation}(V)$ 
3: for  $k \in o$  do
4:   search in  $A$  all the neighbours of  $k$ ,  $j$ 
5:    $\forall j$ , calculate  $\Delta Q_k(j)$  in matrix  $\mathcal{M}$ 
6:    $j^* = \{ j \mid \Delta Q_k(j^*) = \max_j \{ Q_k(j) \} \}$ 
7:   if  $\Delta Q_k(j^*) > 0$  then
8:     Move node  $k$  to  $j^*$  's community
9:   else
10:     $k$  remains in its community
11:   end if
12: end for

```

3.4.1. Código

El código 3.1 es un ejemplo en Python.

En el algoritmo 1 aparece un ejemplo en pseudocódigo.

4

Experimentos / Validación

En esta sección se describen en detalle las queries realizadas sobre este dataset.

4.1. Consultas Realizadas

4.1.1. Descripción General

Piloto más consistente de la temporada 2012

En esta query intentaremos averiguar cuál ha sido el piloto más consistente de la temporada 2012. Ya que este término puede resultar ambigüo, en concreto intentaremos averiguar qué piloto tuvo una menor diferencia entre la media de sus vueltas rápidas del campeonato y la media de todas las vueltas de todos los Grandes Premios de la temporada.

Necesitaremos cruzar varias fuentes de datos para esto:

- `rices.csv`
- `lap_times.csv`
- `drivers.csv`
- `results.csv`

Para filtrar cualquier query por temporada o por rango de temporadas, tenemos que filtrar la tabla `rices` según la columna `year`. En nuestro caso, filtramos la tabla para la temporada 2012.

Una vez tenemos todas las carreras de la temporada en cuestión, necesitamos hallar el tiempo medio por vuelta de cada piloto en cada Gran Premio. En la tabla `lap_times.csv` existe una columna llamada `milliseconds`, que denota los milisegundos de la vuelta dada. Haciendo un `right join`, podemos obtener todas las vueltas dadas en una temporada. Usando una ventana que particione los datos por piloto, haremos la media de la columna `milliseconds` para hallar la media de toda la temporada.

Para obtener la media de las vueltas rápidas de cada piloto, debemos usar la tabla `results`, que tiene una entrada por cada piloto y Gran Premio que contiene además su vuelta más rápida según el formato `MM:ss.mmm`, siendo `MM` los minutos, `ss` los segundos y `mmm` los milisegundos. Para convertir este formato a milisegundos y viceversa, he creado dos UDFs llamadas `lapTimeToMs` y `msToLapTime`.

```
val lapTimeToMs = (time: String) => {
  val regex =
    """([0-9]|[0-9][0-9]):([0-9][0-9])\.([0-9][0-9][0-9])""".r
  time match {
    case regex(min,sec,ms) => min.toInt * 60 * 1000 + sec.toInt * 1000
      + ms.toInt
    case "\\N" => 180000
  }
}: Long
```

```
val msToLapTime = (time: Long) => {  
  val mins = time / 60000  
  val secs = (time - mins*60000)/1000  
  val ms = time - mins*60000 - secs*1000  
  
  val formattedSecs = if((secs / 10).toInt == 0) "0" + secs else secs  
  // if ms = 00x -> "0"+"0"+x . if ms = 0xx -> "0"+ms  
  val formattedMs = if((ms / 100).toInt == 0) "0" + (if((ms /  
    10).toInt == 0) "0" + ms else ms) else ms  
  mins + ":" + formattedSecs + "." + formattedMs  
}: String
```

La función `lapTimeToMs` debe tener en cuenta también que hay entradas que no tienen un tiempo válido, sino que contienen el string `"\N"`. Esto se debe a que hay ocasiones en los que un piloto no termina la vuelta. Un ejemplo sería si se retirase de la carrera en esa vuelta. Se ha decidido sumar 3 minutos (180000 ms) como penalización.

Una vez tenemos estas dos funciones, ya podemos el tiempo medio de vueltas rápidas. De nuevo particionamos los datos según el piloto y hacemos la media de la columna `fastestLapTime`. Este valor en milisegundos se convierte al formato de tiempo de vuelta con la UDF correspondiente y tras esto se ordena por el diferencial hallado de forma ascendente.

Además de esto, haciendo un análisis previo se observó que había pilotos que habían completado pocas carreras en comparación al resto, ya que a que sustituían a algún otro piloto debido a algún accidente. Se decidió filtrar estos outliers de forma que, independientemente del periodo de tiempo sobre el que queramos lanzar esta query, el resultado fuese significativo. Para ello se usó la tabla obtenida con todas las carreras en la temporada y, a partir de la columna `lap` de la tabla `lap_times` se realizó un conteo de todas las vueltas dadas por cada piloto utilizando, de nuevo, una ventana de datos particionada por piloto.

De este `DataFrame` queremos obtener tanto la cuenta de vueltas por piloto como el numero total de pilotos que han participado en la temporada, que haremos de la siguiente manera:

```
val (distinctDrivers, allLaps) = lapCount  
  .agg(  
    countDistinct("driverID"),  
    count(col("lap"))  
  ).as[(BigInt, BigInt)]  
  .collect()(0)  
  
val avgLapsThisPeriod = allLaps.toInt / distinctDrivers.toInt
```

Habiendo obtenido la media de vueltas por piloto de la temporada, podemos filtrar aquellos cuyo conteo sea inferior.

Solo quedaría formatear los datos de salida. Para ello primero se hace un join con la tabla de pilotos, lo cual nos permite concatenar nombre y apellido de cada uno de los participantes en una nueva columna. En esta caso, como solo nos interesa el piloto y su diferencial, estas serán las únicas columnas que dejaremos en el DataFrame.

Sobre qué piloto fue el más consistente en la temporada 2012, este análisis nos dice lo siguiente:

INSERTAR IMAGEN DE RESULTADO

Lo interesante de esta query es que no solo vale para esta temporada. En el dataset que trato se tiene información de tiempos de vuelta desde el año 1996, así que podemos especificar cualquier conjunto de temporadas, ya sean consecutivas, saltadas o solamente una temporada en concreto.

Veamos cuál ha sido el piloto más consistente desde el año 1996:

INSERTAR IMAGEN DEL RESULTADO DESDE 1996 HASTA LA ACTUALIDAD

Dominio de fabricantes en la década de los 90

Con esta query se pretende hallar qué fabricante ha sido el más dominante en la década de los 90. En concreto intentaremos hallar el número de mundiales ganados y el número de carreras ganadas.

Se usarán las siguientes fuentes de datos:

- `rices.csv`
- `constructor_standings.csv`
- `constructors.csv`

Al igual que en la query anterior, si queremos fijar nuestra atención en un periodo de tiempo, tenemos que hacerlo filtrando la columna `year` de la `rices.csv`. En este caso, necesitamos todas las carreras entre el año 1990 y el año 1999.

Una vez obtenidas todas las carreras de la década, tenemos que obtener la última carrera de cada temporada. Esto es debido a que en `constructor_standings.csv` tenemos la clasificación resultante al final de cada carrera. Para ello, usaremos crearemos una ventana en la que particionaremos los datos por año y que usaremos con la función `max()` sobre la columna `round`, que nos indica el índice de la carrera, es decir, la primera carrera de la temporada tendrá `round == 1` para crear una columna llamada `max` en la que guardaremos

el índice de la última carrera de la temporada. Finalmente, filtraremos los datos para quedarnos con aquellos en los que la columna `round === max`.

Tras esto último, unimos las tabla `constructor_standings.csv` con la recién obtenida para quedarnos con los resultados en las últimas carreras y filtramos según la columna `position === 1` para quedarnos con los ganadores. Teniendo esto, podemos ver también que la columna `wins` nos proporciona el número de victorias de cada escudería en esa temporada, así que, creando una ventana en la que particionemos por fabricante podemos hallar tanto la suma de victorias como el conteo de apariciones de cada una.

Pasando ya a la presentación de los datos, se filtrarían los constructores duplicados y se ordenarían los datos según el total de campeonatos ganados primero y, en caso de empate, por número de victorias. Además, se hace un join con la tabla de constructores para obtener su nombre.

INSERTAR CAPTURA DE PANTALLA

Lo interesante de esta query es que se puede usar para cualquier periodo de tiempo. Podemos averiguar por ejemplo qué fabricante ha sido el más dominante en toda la historia de la competición y qué constructor ha dominado ciertos años concretos.

INSERTAR CAPTURA DE PANTALLA DE DOMINIO HISTORICO

4.2. Análisis de requisitos no funcionales

5

Conclusiones y trabajos futuros

En este capítulo se detallan las conclusiones derivadas del TFG y la propuesta de posibles trabajos futuros.

Las citas del texto Autor [3], Autor [4], Autor [5], Autor [6] y Autor [7] deben ir referenciadas en la bibliografía.

5.1. Texto de relleno

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.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat.

Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetur at, consectetur sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetur a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetur. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetur odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetur eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetur tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam.

Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis

vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam congue neque id dolor.

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Bibliografía

- [1] Autores, “Cómo escribir tfg,” *IEEE Informática*, 3081. [Online]. Available: <http://una.url>
- [2] T. Oetiker, H. Partl, I. Hyna, and E. Schlegl, *The Not So Short Introduction to LaTeX*, June 2007, version 4.22. [Online]. Available: <http://tobi.oetiker.ch/lshort/lshort.pdf>
- [3] M. Giaquinta and S. Hildebrandt, *Calculus of variations II*. Springer Science and Business Media, 2013, vol. 311.
- [4] S. Fortune and C. J. Van Wyk, “Efficient exact arithmetic for computational geometry,” in *Proceedings of the Ninth Annual Symposium on Computational Geometry*, 1993, pp. 163–172.
- [5] S. Fortune, “Voronoi diagrams and delaunay triangulations,” *Computing in Euclidean geometry*, pp. 225–265, 1995.
- [6] J. C. Mitchell, “Social networks,” *Annual review of anthropology*, vol. 3, no. 1, pp. 279–299, 1974.
- [7] C. B. Morrey Jr, *Multiple integrals in the calculus of variations*. Springer Science and Business Media, 2009.

Apéndice



Este es el primer apéndice

A.1. Ejemplo de sección

Sección del apéndice