

# **Memoria**

Blanca Cano Camarero y Iker Villegas Labairu.

18 de octubre de 2022

# Indice de contenidos

<b>Prefacio</b>	<b>3</b>
<b>1 Introducción</b>	<b>4</b>
<b>2 Cálculo de <i>spikes</i></b>	<b>5</b>
2.1 Lectura de los datos . . . . .	5
2.1.1 Descripción . . . . .	5
2.1.2 Requisitos . . . . .	5
2.2 Lectura de los datos . . . . .	6
2.3 Diseño del algoritmo de cálculo de <i>spikes</i> . . . . .	7
<b>3 Summary</b>	<b>8</b>
<b>References</b>	<b>9</b>

# Prefacio

Práctica de la asignatura de Teoría de la Información del máster de Ciencia de Datos de la UAM del curso 2022-2023.

Esta memoria ha sido generada con [Quarto](#) y el lenguaje utilizado ha sido [Python](#).

Las funciones declaradas se encuentra en el directorio **src** y contiene las siguientes:

- `read_data.py`: Lee los datos en formato `csv` y devuelve un `dataframe` con ellos.

los notebooks en formato `.qmd` es la memoria ejecutable y que hace llamadas a tales funciones.

Para poder ejecutar la memoria entera dispone de un **Makefile** cuyas funciones básicas son:

- `make` o `make render` Para renderizar un pdf.
- Visualización de la memoria en html `make preview`.

Para ejecutar alguna casilla concreta puede abrir.

# 1 Introducción

**Añadir descripción de la práctica.**

See Knuth (1984) for additional discussion of literate programming.

## 2 Cálculo de *spikes*

### 2.1 Lectura de los datos

#### 2.1.1 Descripción

Para gestión de la información se utilizará la biblioteca de pandas, no es necesario gestionar la memoria porque las arquitecturas de nuestros ordenadores la manejan sin problemas.

La estructura de los ficheros viene dada en la información de los datos, en el fichero `InformacionFicheros.txt` y en las tres primeras líneas de los mismos (las cuales deberán de ser obviadas para la lectura del fichero).

#### 2.1.2 Requisitos

- Tener las respectivas biblioteca instaladas (pandas, matplotlib y numpy).
- Los datos deben encontrarse en el path indicado en la variable `data_path`.

En el siguiente fragmento de código puede observar la cabecera de los datos:

- El intervalo de muestreo es de `0.1ms`.
- Hay dos canales, una por cada neurona.
- Y en total se han tomado 19847700 muestreos.

```
print('Datos fichero trozo C')
print(23*'-')
!head -n 14 ./DatosSinapsisArtificial/InformacionFicheros.txt
```

```
Datos fichero trozo C
-----
```

La estructura de los ficheros es:

```
-----
```

TrozoC.txt -> Control  
Las tres primeras líneas del fichero son:  
Sample interval = 0,100000 ms  
Number of channels = 2  
Number of samples per channel = 19847700

y a continuación las columnas:  
Columna 1 -> LP  
Columna 2 -> VD  
-----

```
## Data information
sample_interval = 0.1
samples_per_channel_trozoC = 19847700
```

## 2.2 Lectura de los datos

Para la lectura de los datos se va a utilizar la biblioteca *Pandas* y la función `read_csv` puede encontrar la implementación de la misma en el directorio `src/read_data.py`.

```
from src.read_data import read_data

data_path = './DatosSinapsisArtificial/'
fileTrozoC, fileTrozoG, fileTrozoR = map(
    lambda letra: data_path+'Trozo'+ letra + '.zip',
    "C G R".split()
)
columns_names = ["LP", "VD"]
#["LP", "VD", "GABAIInjection"]
trozoC = read_data(fileTrozoC, columns_names)
trozoC.head(4)
```

	LP	VD
0	0.004883	0.015259
1	0.001526	0.024109
2	-0.010681	0.031128
3	-0.022278	0.041809

Figura 1: Primeras 4 filas de la señal leída.

## 2.3 Diseño del algoritmo de cálculo de *spikes*

## 3 Summary

In summary, this book has no content whatsoever.



## References

Knuth, Donald E. 1984. «Literate Programming». *Comput. J.* 27 (2): 97-111. <https://doi.org/10.1093/comjnl/27.2.97>.