

Universidad
Rey Juan Carlos

Escuela Técnica Superior
de Ingeniería Informática

Grado en Ingeniería del Software

Curso 2023-2024

Trabajo Fin de Grado

**MÉTODOS DE APRENDIZAJE AUTOMÁTICO
PARA LA CLASIFICACIÓN DE
COMPORTAMIENTOS ESTEREOTIPADOS**

Autor: Gonzalo Ortega Carpintero
Tutor académico: Alberto Fernández Gil
Tutora CSIC: Paula Peixoto Moledo

Agradecimientos

(Por redactar)

Resumen

(Por redactar)

Palabras clave:

- Python
- Neurociencia
- Aprendizaje automático
- ...

Índice de contenidos

Índice de tablas	X
Índice de figuras	XII
Índice de códigos	XIV
1. Introducción	1
1.1. Bibliotecas y herramientas utilizadas	2
2. Objetivos	4
3. Fundamentos teóricos	5
3.1. Métodos no supervisados	5
3.1.1. Agrupamiento por K-medias	6
3.1.2. Agrupación aglomerada	6
3.1.3. Agrupación por afinidad	6
3.1.4. Análisis de componentes principales	6
3.2. Métodos supervisados	6
3.2.1. Red neuronal	6
4. Análisis y experimentación	7
4.1. Preprocesado de datos	7
4.1.1. DeepLabCut	7
4.1.2. Filtrado e interpolación	9
4.1.3. Computo de variables a analizar	10
4.2. Procesado de datos	11
4.2.1. Matriz de similitud	11
4.2.2. Agrupamiento por afinidad	11
4.2.3. Reducción de dimensionalidad para visualización	11
4.2.4. Entrenamiento de una red neuronal	11
5. Resultados	12
6. Conclusiones	13

Bibliografía	13
Apéndices	16
A. Este es el primer apéndice	18
A.1. Ejemplo de sección	18

Índice de tablas

4.1. Extracto del DataFrame de Pandas de los datos sin procesar de DeepLabCut.	8
---	---

Índice de figuras

4.1. Salida de DeepLabCut.	7
4.2. Triangulaciones de los puntos del animal 4128 el 02-12-2020, 1:00:37.	8
4.3. Trayectoria de un punto de cabeza del animal 4128 el 02-12-2020.	9
4.4. Trayectoria filtrada de un punto de la cabeza del animal.	9
4.5. Trayectoria interpolada de un punto de la cabeza del animal.	10

Índice de códigos

1

Introducción

La Neurociencia es la disciplina científica que estudia el sistema nervioso y todas sus componentes. Para el estudio del funcionamiento del cerebro pueden utilizarse multitud de técnicas diferentes, pero para evitar la intrusión en el sujeto de estudio de muchas de ellas, muchos experimentos e investigaciones se basan en el estudio del comportamiento. Es la etología la encargada de este tipo de estudio en animales y en la cual, históricamente, la categorización de comportamientos dependía fuertemente de la observación manual. Eso ha conllevado siempre una excesiva cantidad de tiempo y las desventajas de los posibles errores de percepción humanos, limitando la cantidad y la calidad de los análisis a realizar.

Gracias a los recientes avances en algoritmos de aprendizaje automático, hoy en día es posible computar multitud de datos de forma simultánea, agilizando y automatizando análisis como el ilustrado previamente.

Este Trabajo de Fin de Grado se ha realizado a lo largo de una estancia de prácticas académicas en el Jercog's Team, un equipo de investigación perteneciente al Instituto Cajal, instituto de Neurociencia del Congreso Superior de Investigaciones Científicas. El equipo está centrado en el estudio de la memoria mediante experimentos con ratones, y uno de sus proyectos abiertos consistía en poder elaborar una herramienta para clasificar automáticamente los comportamientos estereotipados de los ratones. Estos son comportamientos cortos, repetitivos y con cierta tendencia a generar patrones, tales como rascarse, caminar en círculos o levantarse a dos patas, ejecutados sin ninguna finalidad, y usualmente inducidos por estar en entornos cerrados y artificiales.

Durante la estancia se han valorado multitud de técnicas de análisis y procesamiento de datos, haciendo hincapié en métodos de aprendizaje automático para

tratar de completar el proyecto gracias a los últimos avances en computación.

1.1. Bibliotecas y herramientas utilizadas

DeepLabCut

DeepLabCut [1] es una herramienta para la estimación 2D y 3D sin marcadores mediante el uso de redes neuronales. Es capaz de identificar y rastrear diferentes partes del cuerpo de múltiples especies realizando todo tipo de comportamientos. Esta ha sido la base de todo nuestro trabajo, ya que todos los videos que hemos analizado han sido procesados en primer lugar por DeepLabCut para rastrear las posiciones de múltiples puntos de los animales a lo largo de los videos de las sesiones. Debido a su importancia, damos una explicación más en detalle de su funcionamiento y de como lo hemos utilizado en la sección 4.1.1.

Google Colab

Google Colab es una herramienta para realizar cuadernos de Jupyter en línea, y poder ejecutarlos en el *backend* de Google. Estos son documentos que intercalan fragmentos de texto con fragmentos de código ejecutable en Python, así como la salida de las distintas ejecuciones. Todo el código realizado para este trabajo ha sido realizado en cuadernos de Colab, y puede ser consultado en los siguientes enlaces:

Análisis principal - <https://colab.research.google.com/drive/1ak2VpDi-zTnV-uEDp-viEkpa8GFDGuBR?usp=sharing>

Scikit-learn

Scikit-learn [2] es una biblioteca de código abierto de Python que contiene numerosas implementaciones de algoritmos de aprendizaje automático. Hemos usado dichas implementaciones tanto como para la explicación teórica de los algoritmos, como para el análisis real de los datos de los animales.

pandas

pandas es una biblioteca de manejo de datos mediante tablas denominadas `DataFrame`. Todos los datos que hemos cargado para ser analizados los hemos

guardado como DataFrames para poder tener un fácil acceso a todos ellos y a sus variables pudiendo, además, visualizarlos de una forma sencilla.

NumPy

NumPy es una de las bibliotecas principales de cálculo científico en Python. Proporciona un objeto de **array** multidimensional y numerosas funciones estadísticas y algebraicas de mucha utilidad.

matplotlib

matplotlib es una biblioteca para crear figuras en Python. Todas las figuras de representación de datos que aparecen en este trabajo han sido creadas con **matplotlib**. El código concreto utilizado para generarlas puede consultarse en los cuadernos de Colab.

2

Objetivos

(Provisional)

El objetivo de este trabajo es hacer una breve introducción a ciertas técnicas de aprendizaje automático aplicables al procesamiento de datos en el análisis de experimentos con animales. Para ello el trabajo se divide en dos capítulos principales.

En el primero abordaremos el estudio teórico de ciertos algoritmos de agrupamiento, reducción de dimensionalidad y de clasificación supervisada. Todo ello desde un punto de vista teórico con ejemplos fáciles de visualizar para ayudar al lector a comprender en profundidad los métodos.

En el segundo capítulo, desarrollaremos paso a paso el análisis real ejercido durante la estancia en el Instituto Cajal. El principal objetivo durante estos meses ha sido buscar la mejor forma de, dados una serie de videos de ratones en su caja, detectar automáticamente diferentes tipos de comportamientos. El fin último del proyecto es estudiar la posibilidad de clasificar dos grupos de animales, control y medicados, basándose en los comportamientos estereotipados.

Para el preprocesado de los videos se ha utilizado la herramienta DeepLabCut, que mediante el uso de redes neuronales y una relativamente pequeña muestra de entrenamiento computa una estimación de la posición de los animales en cada uno de los fotogramas. Tras el preprocesado haremos una revisión de los artículos científicos que han motivado las diferentes técnicas de agrupamiento que hemos acabado utilizando, y finalizaremos desarrollando cada una de ellas a la par que desglosando el código en Python utilizado en cada apartado.

3

Fundamentos teóricos

3.1. Métodos no supervisados

Los algoritmos de aprendizaje automático no supervisados se utilizan cuando no se conoce la salida esperada. Al algoritmo de aprendizaje se le otorgan los datos de entrada y se le pide extraer información de estos datos. Las principales aplicaciones de estos algoritmos, las cuales vamos a aprovechar, son la agrupación de datos y la reducción de dimensionalidad de las variables de los mismos. Esa última es usada principalmente para poder hacer representaciones de datos multidimensionales, los cuales serían complejos de visualizar de otra forma.

La principal pega que pueden tener estos algoritmos es que, si bien no siempre son capaces de identificar conocimiento dados los datos utilizados, cuando lo obtienen, no siempre es el conocimiento que esperábamos obtener. Póngase el ejemplo de un algoritmo que tratase de agrupar rostros de personas iguales. Al no darle a priori ningún tipo de salida de ejemplo, el algoritmo puede acabar clasificando si los rostros están de frente o de lado, no precisamente lo que esperábamos. Es por ello que estos algoritmos cuentan con diversidad de parámetros para ajustarlos a nuestras necesidades, tratando de realizar la agrupación deseada.

En esta sección vamos a estudiar a fondo tres tipos de algoritmos de agrupación: el agrupamiento por k-medias, la agrupación aglomerada, y la agrupación por afinidad. Además, estudiaremos también el principal algoritmo de reducción de dimensiones, el análisis de componentes principales, PCA, de sus siglas en inglés. Los principales ejemplos y explicaciones de los algoritmos han sido inspirados

por los datos en el libro [3, Introduction to Machine Learning with Python].

3.1.1. Agrupamiento por K-medias

3.1.2. Agrupación aglomerada

3.1.3. Agrupación por afinidad

3.1.4. Análisis de componentes principales

3.2. Métodos supervisados

[4, Deep Learning with PyTorch]

3.2.1. Red neuronal

4

Análisis y experimentación

4.1. Preprocesado de datos

4.1.1. DeepLabCut

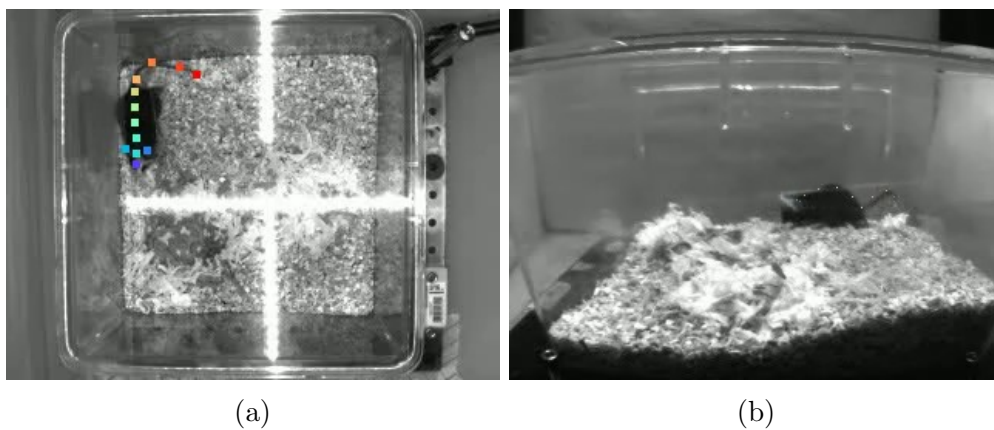
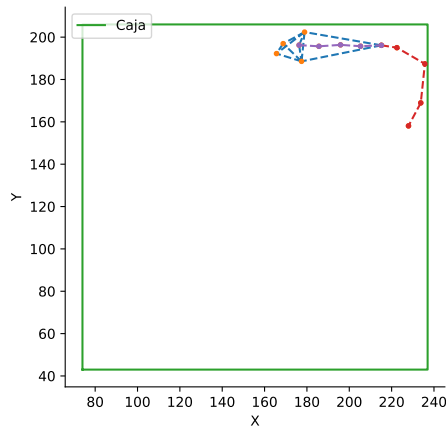


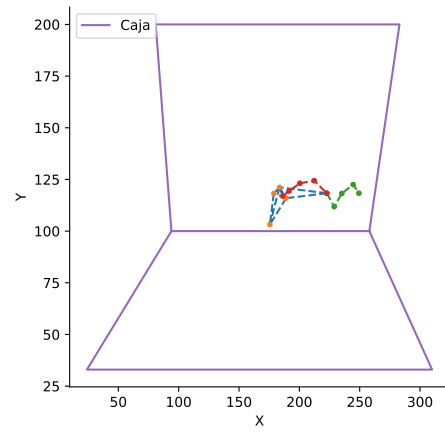
Figura 4.1: Salida de DeepLabCut del animal 4128 el 02-12-2020, 1:00:37. [4.1a](#) Video de la vista cenital de la caja. Los puntos sobre el animal son los dibujados por DeepCutLab para rastrear las partes del animal. [4.1b](#) Video de la vista lateral de la caja.

	Nosex	Nosey	Noselikelihood	Headx	Heady	...
0	136.165344	177.722496	0.000084	129.790253	174.772552	...
1	162.032005	201.444756	0.942181	168.152061	202.420639	...
2	156.297043	200.326378	0.000073	162.436028	203.156837	...
3	155.370415	199.043297	0.000277	159.507599	200.199928	...
4	149.272644	197.677170	0.000045	155.493912	198.814835	...
...

Tabla 4.1: Extracto del DataFrame de Pandas de los datos sin procesar de DeepLabCut.



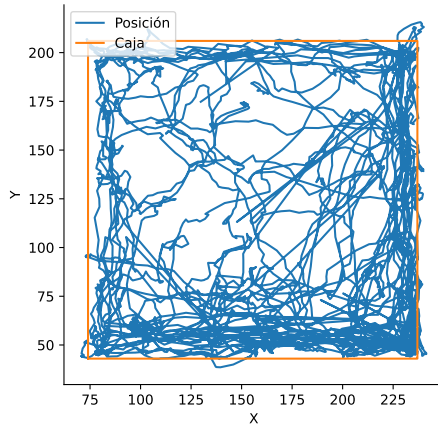
(a) Vista cenital.



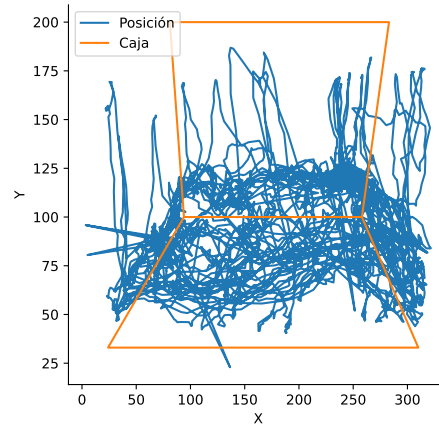
(b) Vista lateral.

Figura 4.2: Triangulaciones de los puntos del animal 4128 el 02-12-2020, 1:00:37.

4.1.2. Filtrado e interpolación

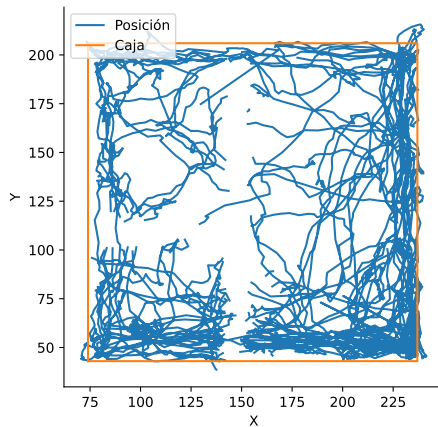


(a) Vista cenital.

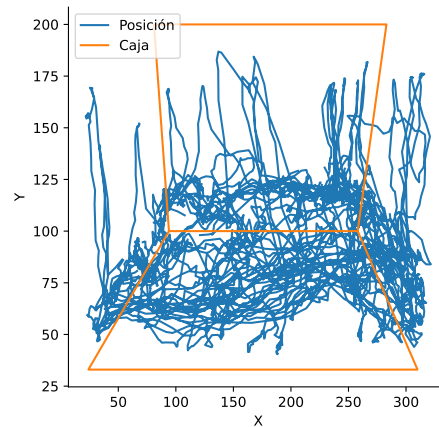


(b) Vista lateral.

Figura 4.3: Trayectoria de un punto de cabeza del animal 4128 el 02-12-2020.

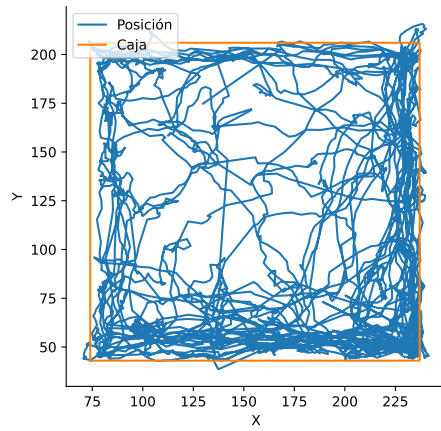


(a) Vista cenital.

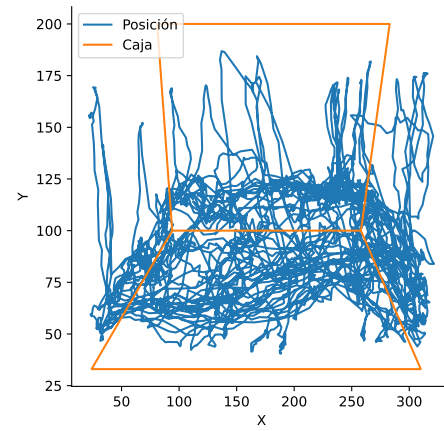


(b) Vista lateral.

Figura 4.4: Trayectoria filtrada de un punto de la cabeza del animal.



(a) Vista cenital.



(b) Vista lateral.

Figura 4.5: Trayectoria interpolada de un punto de la cabeza del animal.

4.1.3. Computo de variables a analizar

4.2. Procesado de datos

4.2.1. Matriz de similitud

4.2.2. Agrupamiento por afinidad

4.2.3. Reducción de dimensionalidad para visualización

4.2.4. Entrenamiento de una red neuronal

5

Resultados

(Por redactar)

6

Conclusiones

(Por redactar)

Bibliografía

- [1] A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, and M. Bethge, “Deeplabcut: markerless pose estimation of user-defined body parts with deep learning,” *Nature Neuroscience*, 2018. [Online]. Available: <https://doi.org/10.1038/s41593-018-0209-y>
- [2] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [3] A. C. Müller and S. Guido, *Introduction to Machine Learning with Python*. O’Reilly, 2017.
- [4] E. Stevens, L. Antiga, and T. Viehmann, *Deep Learning with PyTorch*. Manning, 2020.

Apéndices



Este es el primer apéndice

A.1. Ejemplo de sección

Sección del apéndice