

TESIS
MAESTRÍA EN CIENCIAS FÍSICAS

**ANÁLISIS NO SUPERVISADO DEL
COMPORTAMIENTO ANIMAL DURANTE EL
APRENDIZAJE MOTOR**

Orientación: Física en Medicina y Biología

Lic. Álvaro Concha

Dra. Soledad Espósito
DIRECTORA

Dr. Damián Hernández
CO-DIRECTOR

Lic. Álvaro Concha
MAESTRANDO

Isntituto Balseiro
Comisión Nacional de Energía Atómica
Universidad Nacional de Cuyo

S. C. de Bariloche, Febrero de 2022

Índice de contenidos

Resumen	xi
Abstract	xiii
Introducción	1
0.1. Motivación	1
0.2. Objetivos	1
1. Caracterización y clasificación del comportamiento	5
1.1. Protocolo experimental	5
1.2. Análisis de video	6
1.3. Algoritmo t-SNE	8
1.3.1. Introducción y formalismo	8
1.3.2. Cálculo de similaridades	9
1.3.3. Función de costo	11
1.3.4. Obtención del mapa de comportamientos	13
1.4. Transformación <i>watershed</i>	14
2. Análisis e interpretación del mapa de comportamientos	17
2.1. Interpretación del mapa	17
2.2. Representatividad de la clasificación	19
2.3. Relación entre el comportamiento y la velocidad del <i>rotarod</i>	21
2.4. Dinámica temporal de transiciones entre poses	23
3. Registros de actividad neuronal	27
3.1. Registro de actividad de neuronas individuales	27
3.2. Actividad de neuronas individuales	28
3.3. Patrones de actividad de grupos de neuronas	30
Conclusiones	37
Bibliografía	41
A. Implementación y detalles técnicos del algoritmo t-SNE	43
A.1. Métodos aproximados	43
A.2. Técnicas de optimización	44
A.3. Parámetros utilizados en el mapa de comportamientos	45
B. Patrones de actividad de grupos de neuronas, continuación	47

Índice de figuras

1.	El estudio de las correlaciones entre la actividad neuronal y los comportamientos animales suscitados, puede ayudar a dilucidar el funcionamiento de diferentes regiones del sistema nervioso.	2
1.1.	Esquema del aparato rotarod.	6
1.2.	Caracterización de las posiciones de partes del cuerpo del ratón en función del tiempo. (a) Fotograma típico con las posiciones de los marcadores de la base de la cola y de las patas traseras (indicados por círculos). A partir de estas posiciones se calculan posiciones relativas (indicadas por flechas). (b) Serie temporal típica que muestra el movimiento sobre el eje vertical de los marcadores de la base de la cola y de las patas traseras (línea continua) y sus valores promedio en este intervalo de tiempo (línea discontinua). Con fines de visualización, los valores promedio de las series temporales se fijaron a 1 cm entre sí. Se marcó una instancia de movimiento activo (flecha en línea continua negra) y de movimiento pasivo (flecha en línea discontinua negra).	8
1.3.	Etograma obtenido a partir de la transformada t-SNE del conjunto de vectores de estado $\{\mathbf{x}_i\}$ de todos los <i>frames</i> de video analizado. Estos vectores de estado representan las posiciones relativas de algunas partes del cuerpo de los ratones. Dentro del recuadro se muestra más en detalle los puntos de una región del etograma.	13
1.4.	Estimación de la distribución de densidad de probabilidad con <i>kernel</i> gaussiano (KDE) de puntos del mapa t-SNE de la Figura 1.3. (a) Superficies de nivel de la KDE. (b) Gradiente de la KDE.	14
1.5.	Mapa de comportamientos obtenido a partir del conjunto de vectores de estado $\{\mathbf{x}_i\}$ de todos los fotogramas de video analizados. En este mapa t-SNE se anotaron los números de clase (<i>labels</i>) asignados a cada región segmentada. Estas anotaciones se posicionaron encima de los centros de masa de cada región. Se marcaron las fronteras entre regiones contiguas. Los puntos están coloreados de acuerdo al <i>label</i>	16
2.1.	Fotogramas típicos extraídos de los videos de los ratones expertos durante la realización de la tarea <i>rotarod</i> . Se extrajo un fotograma representativo por cada clase de <i>label</i> , obtenido a partir de la segmentación del mapa de comportamientos de la Figura 1.5.	18
2.2.	Serie temporal típica de <i>labels</i> . En la parte superior se muestran la evolución temporal de las coordenadas verticales de las patas izquierda y derecha del ratón. Se indican con círculos los máximos de las posiciones de las patas izquierda y derecha y se trazan flechas hacia los correspondientes <i>labels</i> 7 y 2.	19
2.3.	Puntos correspondientes a cada ratón por separado en el etograma obtenido.	19

2.4. Caracterización de la estadística de las secuencias de <i>labels</i> y su dependencia con la velocidad del <i>rotarod</i> . (a) Duración promedio de cada <i>label</i> en la secuencia, con sus intervalos de confianza. (b) Probabilidad de ocurrencia de cada <i>label</i> en la secuencia, bajo tres regímenes de velocidad. Las barras de error representan la desviación estándar del promedio de las probabilidades al estudiar los <i>trials rotarod</i> por separado. Las probabilidades para cada ratón por separado se indican con marcadores vacíos de color negro (círculos: ratón 1, triángulos: ratón 2 y cuadrados: ratón 3).	22
2.5. Transiciones a primer orden en la secuencia de <i>labels</i> . (a) Grafo de transiciones. El grosor de las flechas es proporcional a la probabilidad de transición. Las flechas en línea continua negra representan transiciones que parten desde un <i>label</i> con probabilidad marginal mayor a $1/N_{\text{clases}} = 0.11$ en la secuencia (para ver las probabilidades marginales de cada <i>label</i> referirse a la Figura 2.4b). Lo opuesto es cierto para las flechas en línea discontinua gris. Solamente se graficaron transiciones que ocurren con probabilidad mayor a $1/(N_{\text{clases}} - 1) = 0.125$. (b) <i>Heatmap</i> indicando las variaciones en la probabilidad de transición a altas velocidades respecto de bajas velocidades.	23
2.6. Logos de secuencias condicionales, centrados alrededor de cada una de las clases de <i>label</i> . La altura de cada <i>label</i> en una determinada columna en la posición relativa k es proporcional a la probabilidad de transición de orden k relativa al <i>label</i> central.	26
3.1. Esquema de la ubicación del arreglo de electrodos para el registro electrofisiológico de neuronas en el MLR.	28
3.2. Diferentes patrones de actividad neuronal registrados alrededor de un tipo de evento de transición entre poses clasificadas (en este caso, el evento es una transición hacia el <i>label</i> 2). Tasas de disparo normalizadas y <i>raster plots</i> para tres neuronas diferentes, cada una durante una prueba <i>rotarod</i> distinta. Los intervalos de error de las tasas de disparo se obtuvieron mediante <i>bootstrapping</i>	29
3.3. <i>Heatmaps</i> con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al <i>label</i> 2. Cada fila de un <i>heatmap</i> representa el <i>z-score</i> de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.	32
3.4. <i>Heatmaps</i> con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al <i>label</i> 5. Cada fila de un <i>heatmap</i> representa el <i>z-score</i> de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.	33
3.5. <i>Heatmaps</i> con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al <i>label</i> 6. Cada fila de un <i>heatmap</i> representa el <i>z-score</i> de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.	34
3.6. <i>Heatmaps</i> con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al <i>label</i> 7. Cada fila de un <i>heatmap</i> representa el <i>z-score</i> de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.	35
A.1. Métodos aproximados para el cálculo de interacciones entre puntos. (a) Partición obtenida mediante un árbol cuaternario (<i>quadtree</i>) (Aproximación Barnes-Hut). (b) Grilla de interpolación (puntos rojos equiespaciados en el plano).	44

Índice de tablas

1.1.	Elementos básicos de la transformada t-SNE y notación utilizada. Comparación entre los espacios de entrada y de salida.	9
2.1.	Frecuencia condicional $p_{l r}$ de observar el <i>label</i> l del etograma de la Fig. 1.5 con el r -ésimo ratón. Valores de diversidad 2D_l calculados para cada <i>label</i> l . Se resaltan en negrita aquellos valores de diversidad mayores a ${}^2D_l = 2.5$. . .	21
A.1.	Parámetros utilizados durante la ejecución del algoritmo t-SNE mediante el cual se obtuvo el etograma de la Figura 1.3.	46

Resumen

Un supuesto fundamental en la neurociencia del comportamiento es que la actividad animal, durante la ejecución de una tarea definida, puede ser descrita en gran parte mediante un conjunto finito de comportamientos estereotipados [1]. A su vez, la información acerca del comportamiento animal puede correlacionarse con registros simultáneos de la actividad neuronal con el fin de estudiar cómo el cerebro codifica diferentes movimientos, cuáles son los circuitos neuronales subyacentes y cómo estos se modifican durante el aprendizaje de nuevas tareas motoras [2, 3].

Sin embargo, la clasificación en categorías de distintos tipos de comportamientos puede ser un proceso complejo. Por un lado, la extensión de los registros de actividad animal puede ser demasiado grande como para clasificar movimientos manualmente y dicha clasificación manual puede ser poco reproducible entre sujetos. Por otro lado, las categorías creadas heurísticamente (por ejemplo, caminar, correr, saltar) suelen ignorar información inherente a la variabilidad intra- e inter-animal típica de los comportamientos no restringidos [1].

Por lo tanto, en este trabajo se utilizaron técnicas de aprendizaje automático no supervisado para clasificar comportamientos exhibidos por ratones expertos durante la ejecución de una tarea motora. En particular, se utilizó el algoritmo t-SNE para clasificar el comportamiento animal, a partir de un conjunto de datos con información acerca de las posiciones de los ratones y de sus partes del cuerpo mientras caminaban sobre un cilindro que gira a velocidades crecientes (tarea *rotarod* con aceleración) [1, 4, 5].

Aplicando dicho algoritmo logramos clasificar el comportamiento en 9 clases individuales que corresponden a poses específicas del ratón, adoptadas mientras realiza la tarea de *rotarod* con aceleración. Estas poses son representativas del conjunto de ratones observados ($N_{ratones} = 3$) y capturan los aspectos comunes de sus movimientos. Además, se estudió la dinámica de transiciones entre poses y la dependencia de las mismas con la velocidad del cilindro *rotarod*. Este análisis nos permitió demostrar que al aumentar la velocidad del cilindro los animales cambian su estrategia motora seleccionando un tipo de locomoción caracterizada por la alternancia entre los miembros inferiores.

Finalmente se utilizó esta clasificación de poses para estudiar posibles correlaciones con patrones de actividad neuronal en la región locomotora del mesencéfalo (MLR) y mostramos la existencia de neuronas cuya actividad está modulada por eventos de transición entre poses. Este trabajo demuestra el beneficio del uso de estrategias no supervisadas para el estudio detallado del comportamiento animal y su utilidad en el estudio funcional de los circuitos neuronales.

Palabras clave: ROTAROD, COMPORTAMIENTO ANIMAL, UMAP, APRENDIZAJE AUTOMÁTICO, CLASIFICACIÓN NO SUPERVISADA, INTERPRETABILIDAD DE MODELOS

Abstract

UNSUPERVISED ANALYSIS OF ANIMAL BEHAVIOR DURING MOTOR LEARNING

A fundamental assumption in behavioral neuroscience is that animal activity, while performing defined tasks, can be vastly described by a finite set of stereotyped behaviors. Information about animal behavior can then be correlated with simultaneous recordings of neural activity allowing us to understand how the brain encodes particular behaviors, what are the underlying neural circuits and how these circuits are modified during motor learning [2, 3].

However, classifying different types of movements can be a complex endeavor. On the one hand, the extent of animal activity recordings may be too large to be manually classified and such classification may not be reproducible between subjects. On the other, heuristically created categories (e.g., walking, running, jumping) tend to ignore inherent information regarding intra- and inter-animal variability frequently found in unrestrained behavior [1].

Therefore, in this work we used unsupervised machine learning techniques to classify different types of behaviors exhibited by expert mice performing a motor skill task. In particular, we used the t-SNE algorithm to classify animal behavior from a dataset containing information about the position of the mouse and their body parts while walking on a rotating cylinder at increasing speeds (accelerating rotarod task) [1, 4, 5].

By applying this algorithm, we were able to classify behavior into 9 individual classes that correspond to specific poses of the mouse while performing the accelerating rotarod task. These poses were found to be representative of the individual mice analyzed ($N_{\text{ratones}} = 3$) and capture common aspects in their movements. In addition, we studied the dynamics of transitions between poses and their dependence with the rotarod speed. This analysis showed that at faster rotarod speed animals change their motor strategy by selecting a type of locomotion characterized by hindlimb alternation.

Finally, this pose classification was used to study potential correlations with neural activity patterns in the mesencephalic locomotor region (MLR) and we found the existence of individual neurons whose activity is modulated by the occurrence of pose transition events. Altogether, this work demonstrates the benefit of the use of unsupervised strategies for the detailed study of animal behavior and its utility in the study of neural circuit function.

Keywords: ROTAROD, ANIMAL BEHAVIOR, UMAP, MACHINE LEARNING, UNSUPERVISED CLASSIFICATION, MODEL INTERPRETABILITY

Introducción

0.1. Motivación

Espero que este trabajo sirva para documentar el *pipeline* de procesamiento de datos.

0.2. Objetivos

Esperamos que este trabajo sirva como documentación del *pipeline* de procesamiento de datos, refinado luego de un año de ajustes durante el transcurso de la maestría.

Más importante aún para los intereses del laboratorio, esperamos que estas herramientas de procesamiento puedan no solamente generalizarse a nuevos experimentos o modelos de comportamiento animal, sino también brindar claridad sobre los resultados.

Un problema común en las aplicaciones que involucran el análisis de grandes volúmenes de datos es su difícil interpretabilidad. Por ejemplo, los modelos de aprendizaje automático basados en redes neuronales profundas pueden generar predicciones muy precisas, sin embargo suele ser difícil interpretar el funcionamiento del modelo [cita requerida]. De manera análoga, los mapeos no lineales son una herramienta popularizada recientemente en diversos ámbitos, desde el análisis de secuenciamiento de células individuales en bioinformática, pasando por el estudio de similaridad de textos en el análisis de lenguaje de natural, hasta sistemas de recomendación de productos usados en la industria de la información [citas requeridas].

En este sentido, la etología, ciencia que estudia el comportamiento animal, no es una excepción. Especialmente cuando abundan los registros de video de comportamiento animal, gracias al fácil acceso a cámaras digitales y su versatilidad para ser incluidas en la mayoría de los protocolos experimentales existentes. En simultáneo, la aparición de herramientas de *software* para el trackeo de movimientos sin marcadores físicos, abrieron la puerta a una cantidad impensada de datos crudos. Es en este contexto de abundancia de información y escasez de métricas comportamentales detalladas, exhaustivas y cuantitativas que entran en juego los mapeos no lineales.

Aquí es donde podría entrar el rotarod y la latencia a caer
Clásicamente

Mencionar *feature discovery* ->behavior labels from UMAP segmentation

Mencionar de nuevo (probablemente lo hice en la tesis) la necesidad de equiparar la cuantificación del comportamiento animal, con el aumento en la precisión y la mejora en los registros de actividad neuronal.

Además, con este enfoque, estoy analizando la biomecánica del movimiento de los ratones en la tarea rotarod. Lo cual es valioso.

Estoy retrocediendo pasosz "descuartizando" la caja negra del mapeo UMAP y los behavior labels. El primer paso atrás es sintetizar las frecuencias promedio de diferentes grupos de ángulos de marcadores corporales. El segundo paso atrás es calcular las diferencias de fase entre pares de ángulos de marcadores corporales.

Podría agregar otras métricas, como la volatilidad (std en una ventana móvil) de los espectros de frecuencia y de los ángulos.

Estos son los features sintetizados o interpretados. Y funcionan bastante bien con la perspectiva de label sequences averaging.

Por otro lado, puedo usar directamente los features del PCA, espectros de frecuencias y ángulos, para intentar ver qué es lo que significa cada behavior label. Esto funciona mejor con la perspectiva de label time series.

En ambas perspectivas, la idea es entrenar un árbol de decisión, que nos diga qué parte de los features decide cada label, y la importancia de cada label en esta clasificación de comportamientos.

En cuanto a descartar label sequences con duración menor a 30 frames: tiene ciertas ventajas. Por un lado, usar label sequences de mayor duración me ayuda a obtener promedios y correlaciones mejor definidas (más número de muestras por ocurrencia de label). Por otro lado, según el supuesto de estereotipia, un comportamiento x sigue siendo un comportamiento x, no importa si su realización dura 5 o 200 frames. Así que para caracterizar el comportamiento, puedo concentrarme en las ocurrencias de mayor duración, para facilitar la interpretación y la estimación/cálculo de las métricas que elegí para hacerlo.

Dado que estoy elevando el espectro de frecuencias wavelet al cuadrado, lo que estoy calculando power spectral centroid, solamente que haciendo un promedio geométrico porque trabajo con canales de frecuencia de varios órdenes de magnitud diferentes (de 0.5 Hz a 50 Hz, son tres órdenes de magnitud comprendidos).

Marcapasos aleatorio (Random Pacemaker Model)

Agregar frase como subtítulo de esta sección

.^All models are wrong, but some are useful George Box.

Filtros: Median filter to denoise positions and likelihoods before Kalman Kalman filter to reconstruct low likelihood tracking (try to correct occlusions and non-confident (low likelihood) marker switchings) Quantile filter to correct Kalman drift and confident (high likelihood) marker switching Finally, a trimmed mean filter, with a small window, to smooth and denoise the signal a little bit.

En etología, la ciencia que estudia el comportamiento, el supuesto de estereotipia consiste en que los comportamientos exhibidos por un animal pueden ser descompuestos en elementos discretos y reproducibles. Estos comportamientos elementales se suponen constantes a lo largo del tiempo y consistentes entre diferentes individuos y, en algunos casos, entre diferentes especies. Estos conjuntos discretos de comportamientos podrían emerger a partir de diferentes mecanismos, por ejemplo, la existencia de límites mecánicos para el control de la marcha y otros movimientos, la formación de hábitos, o la presión selectiva para ejecutar acciones robustas u óptimas en un determinado entorno. Estos mecanismos limitarían el repertorio de movimientos del animal a pesar de su potencial capacidad de moverse de infinitas maneras, restringidos en principio únicamente por los límites biomecánicos de su morfología [1, 6].

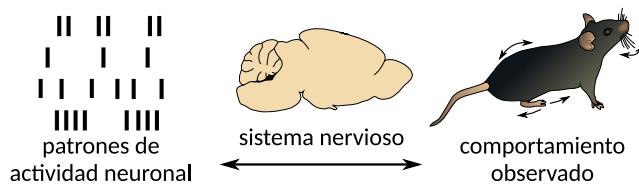


Figura 1: El estudio de las correlaciones entre la actividad neuronal y los comportamientos animales suscitados, puede ayudar a dilucidar el funcionamiento de diferentes regiones del sistema nervioso.

El concepto de estereotipia tiene una gran utilidad cuando se trata de comprender có-

mo el cerebro controla dichos movimientos. En este caso, la información resultante de la cuantificación del comportamiento es de fundamental importancia para su correlación con registros simultáneos de la actividad neuronal con el fin de estudiar cómo el cerebro codifica diferentes movimientos, cuáles son los circuitos neuronales subyacentes y cómo estos se modifican durante el aprendizaje de nuevas tareas motoras (Figura 1) [2, 3]. A pesar de su importancia, la determinación del repertorio de movimientos de un animal resulta experimentalmente complicada, debido a las dificultades que conlleva la cuantificación del comportamiento animal en términos objetivos y reproducibles.

En este trabajo nos proponemos explorar el espacio de comportamientos a los que accede un conjunto de ratones realizando una tarea de destreza motora de manera no supervisada. La tarea estudiada se denomina rotarod con aceleración, la cual consiste en entrenar ratones para que caminen sobre un cilindro que gira a velocidades crecientes, con aceleración constante. El *rotarod* con aceleración es un paradigma experimental ampliamente utilizado en el estudio del aprendizaje motor dado que la *performance* del animal mejora con el entrenamiento de manera dependiente de la plasticidad neuronal [7]. Sin embargo, el desempeño del animal en esta tarea se suele evaluar simplemente a través de la medición del tiempo de permanencia en el cilindro. En este contexto nos preguntamos, ¿cuáles son los cambios en el repertorio de movimientos del animal asociados a un mejor desempeño de la tarea?

Para contestar esta pregunta realizamos una serie de filmaciones de los animales mientras realizaban la tarea motora. Para cuantificar los movimientos, se extrajeron coordenadas bidimensionales de las posiciones de algunas partes de sus cuerpos y su evolución en el tiempo. Para ello, se utilizó *DeepLabCut*, un *software open-source* de *motion-tracking*, que no requiere la utilización de marcadores físicos adicionales sobre los cuerpos de los animales [8]. A partir de estas coordenadas se calcularon posiciones relativas en función del tiempo. De esta manera, se extrajo información cuantitativa de las posiciones de los ratones y de sus partes del cuerpo durante el transcurso de la tarea *rotarod*.

Para dilucidar la estructura intrínseca de este conjunto de datos se utilizó el algoritmo t-SNE para la reducción de la dimensión de manera no lineal [4]. La transformada t-SNE es también una herramienta popular para la visualización de conjuntos de datos de dimensión alta y se utiliza frecuentemente en el ámbito de la bioinformática para visualizar datos provenientes de la secuenciación del material genético de células individuales [9]. Así se construyó un mapa de comportamientos en 2 dimensiones donde se proyectan cada uno de nuestros vectores de dimensión mayor que describen la posición de los ratones y de sus partes del cuerpo.

Luego, procedimos a analizar la dinámica de transiciones en este espacio de dimensión reducida producido por el algoritmo t-SNE y se identificaron *clusters* de puntos que se corresponden con poses específicas adoptadas por los ratones durante la ejecución de la tarea *rotarod*. Así, se utilizó de manera efectiva al algoritmo t-SNE para la clasificación no supervisada de poses animales. Esta clasificación es robusta y representativa de nuestro conjunto de ratones estudiados ($N_{ratones} = 3$) y tiene potencial tanto como para reimplementarse utilizando un mayor número de ratones *a priori*, como también para agregar nuevas observaciones experimentales al mapa t-SNE *a posteriori* [9, 10]. Interpretaremos entonces como comportamientos estereotipados a diferentes sucesiones de estas poses clasificadas. En particular, nos concentraremos en el estudio de transiciones de primer orden entre dos poses distintas, pero también exploraremos la dinámica de transiciones de poses a escalas de tiempo mayores.

En simultáneo, durante la ejecución de la tarea *rotarod*, se registró la actividad de neuronas individuales de la región locomotora del mesencéfalo (MLR) de los ratones ya entrenados en la tarea. El MLR es una región implicada en los procesos de iniciación y de control de la velocidad de la locomoción [11-13]. Por lo tanto, en este experimento nos preguntamos si la actividad de algunas de las neuronas presentes en esta región del cerebro estaría modulada por el movimiento del ratón en el transcurso de la tarea *rotarod* [14-16]. Para contestar

esta pregunta correlacionamos la actividad neuronal con la ocurrencia de transiciones entre poses específicas obtenidas a partir de la clasificación no supervisada del comportamiento. Este estudio nos permitió demostrar la existencia de neuronas en el MLR (alrededor de un cuarto del total de neuronas registradas) cuya actividad neuronal varía significativamente alrededor de las transiciones entre poses clasificadas. Este resultado sugiere fuertemente que el MLR no solo codificaría la iniciación y la velocidad de la locomoción sino también el tipo de movimiento seleccionado.

Capítulo 1

Caracterización y clasificación del comportamiento

Con el objetivo de comprender cómo se modifica el comportamiento animal durante el aprendizaje de una nueva habilidad, decidimos comenzar estudiando el patrón de movimientos de ratones previamente entrenados en la tarea *rotarod* con aceleración. Para ello clasificaremos, de manera no supervisada, diferentes tipos de poses que los ratones adoptan durante la ejecución de la tarea con el objetivo final de cuantificar el comportamiento animal de manera reproducible y automatizada.

A continuación explicaremos el procedimiento llevado a cabo para la clasificación del comportamiento animal. Primero, en la Sección 1.1 describiremos el protocolo experimental de entrenamiento al cual fueron sometidos los ratones. Es importante aclarar que una vez que los ratones fueron entrenados en la tarea *rotarod*, se registraron videos de la actividad de los mismos durante la ejecución de la tarea aprendida. Luego de la descripción del protocolo experimental, detallaremos en la Sección 1.2 el proceso de análisis de los videos de los ratones, con el fin de extraer las posiciones de algunas partes de sus cuerpos en función del tiempo. Después, en la Sección 1.3 explicaremos cómo se clasificaron estos conjuntos de datos de posiciones utilizando un algoritmo de reducción de la dimensión llamado t-SNE. Utilizando el algoritmo t-SNE obtendremos un mapa bidimensional de comportamientos de los ratones. Finalmente, en la Sección 1.4 introduciremos el método utilizado para identificar y segmentar *clusters* en este mapa de comportamientos.

1.1. Protocolo experimental

En primera medida, es importante introducir el protocolo de entrenamiento de los ratones que se siguió en esta parte del trabajo. Este protocolo puede dividirse esencialmente en dos partes, habituación y entrenamiento, y tiene una duración total de entre 7 a 8 días.

1. Habitación (días 1 y 2): antes de dar inicio a las sesiones de entrenamiento (pruebas o *trials*), los animales son habituados al experimentador y al ambiente del experimento (es decir, a la sala donde se encuentra el rotarod y al aparato *rotarod* en sí). Esta etapa comprende los dos primeros días del protocolo, en los cuales los ratones son trasladados a la sala de experimentación y expuestos al *rotarod*, fuera de funcionamiento, durante 10 a 15 minutos.
2. Entrenamiento (día 3 en adelante): cada día de entrenamiento consiste en 4 pruebas sucesivas de *rotarod*, separadas entre sí por 5 minutos de descanso. En cada prueba el animal es colocado sobre el cilindro del *rotarod*, que gira a velocidades crecientes con

aceleración constante, desde 5 rpm hasta 45 rpm en 5 minutos, o hasta que el ratón se cae (Figura 1.1). La etapa de entrenamiento comprende entre 5 y 6 días consecutivos. El tiempo transcurrido desde el inicio de la prueba hasta el momento en el que el ratón se cae del cilindro se denomina latencia a caer.

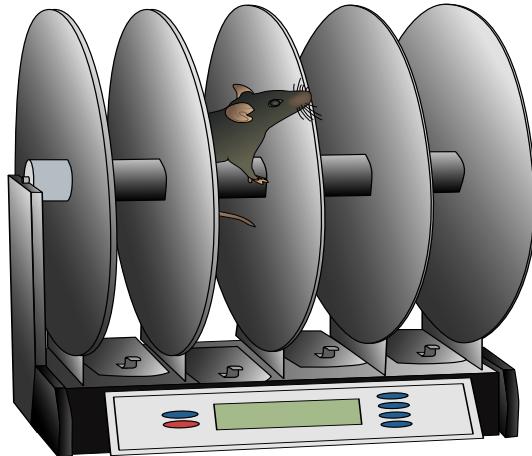


Figura 1.1: Esquema del aparato rotarod.

Cabe mencionar que cada prueba de *rotarod* se lleva a cabo con un único ratón por vez sobre el cilindro, a pesar de que el aparato esté diseñado para albergar a más de un ratón simultáneamente. Esto se debe a que la presencia de múltiples ratones sobre el cilindro influye en sus comportamientos individuales.

Con el objetivo de correlacionar la actividad neuronal con el comportamiento (Capítulo 3), los animales expertos analizados en este proyecto habían sido anteriormente inyectados en la región locomotora del mesencéfalo (MLR) con un virus que expresa canalrodopsina (ChR) sólo en neuronas glutamatérgicas. Se trata de un virus Cre-dependiente inyectado en un ratón transgénico que expresa Cre solamente en neuronas glutamatérgicas. Estos animales fueron también implantados con electrodos, para el registro de la actividad eléctrica en el MLR, y una fibra óptica, que permite la administración de luz para la activación de la ChR. De esta manera, se pretende lograr la identificación de la actividad registrada originada por neuronas glutamatérgicas a partir de su respuesta rápida y fiable a la luz. Cabe aclarar que ninguna de las neuronas registradas fueron positivamente identificadas como glutamatérgicas [7, 17, 18].

Una vez entrenados, los animales que denominaremos “expertos” en la tarea *rotarod* fueron filmados mientras realizaban la tarea motora y se registró en simultáneo su actividad eléctrica en el MLR.

1.2. Análisis de video

En esta parte del trabajo se analizaron los videos de los ratones expertos, adquiridos durante la realización de pruebas *rotarod*. En estos videos se capturaron los movimientos de la parte posterior de los ratones. A partir de estos videos se extrajeron las posiciones *x* e *y* de las extremidades posteriores y de la cola del ratón, en función del tiempo, a una frecuencia de muestreo de 100 Hz. Para ello se utilizó *DeepLabCut*, una herramienta *open-source* que estima la posición de marcadores digitales definidos por el usuario. *DeepLabCut* permite la estimación de poses animales basado en técnicas de transferencia de aprendizaje (*transfer learning*) con redes neuronales artificiales profundas (*deep neural networks*) y puede alcanzar

un rendimiento equivalente al humano luego de haber entrenado la red con pocos fotogramas (*frames*) de ejemplo (típicamente 50-200 *frames*) [8]. Cabe mencionar que en esta parte del trabajo se contó con un total de 59 videos, provenientes de $N_{\text{ratones}} = 3$ ratones diferentes sometidos al menos tres veces por día a pruebas *rotarod*, durante un promedio de 6 días por ratón. En total, el cuerpo de videos adquiridos comprende alrededor de $N \sim 1$ millón de fotogramas para analizar.

La Figura 1.2a muestra un fotograma típico de video analizado. En él, se marcaron las posiciones extraídas para las patas traseras y la base de la cola (con círculos en la figura). Además para cada fotograma se extrajeron las posiciones de las cuatro esquinas del cilindro rotarod. Esto se hizo a modo de calibración ya que se sabe que el ancho del cilindro es de 5.7 cm, permitiéndonos convertir los valores de las posiciones desde píxeles a centímetros. En la práctica, para cada *trial rotarod* capturado en video, se calculó la mediana de la distancia entre las dos esquinas inferiores del cilindro rotarod en píxeles de forma tal de evitar la influencia de valores extremos. Esto sucede cuando un marcador digital es ocultado parcial o totalmente en el video (por ej., si el ratón oculta con su cuerpo a una de las esquinas del cilindro) y como consecuencia, el programa *DeepLabCut* pierde precisión en la determinación de la posición del marcador ocultado.

La Fig. 1.2b muestra la evolución temporal típica de las componentes verticales de las posiciones de los marcadores de las patas traseras y de la base de la cola, en un intervalo de tiempo de 10 s luego de 10 s de comenzada la prueba de *rotarod*. Pueden observarse patrones periódicos de movimiento en los diferentes marcadores y puede notarse que estos patrones de movimiento son asimétricos en el tiempo. También pueden observarse diferentes desfasajes en los movimientos de los marcadores, como así también diferentes grados de sincronización en el movimiento. En particular se observan de manera clara las instancias de movimiento voluntario (activo) e involuntario (pasivo) de las patas del ratón. Por un lado, junto a la serie temporal de la pata derecha en la Figura 1.2b se indica con una flecha en línea continua negra el momento en el que el ratón da un paso con la pata derecha, levantándola bruscamente y desplazándose activamente hacia la parte superior del cilindro. Por otro lado, la flecha en línea discontinua negra indica el movimiento pasivo de deriva debido a la rotación del cilindro.

Finalmente, debemos aclarar que se trabajó con posiciones relativas. Denotemos con \mathbf{T}_i (*tail*), \mathbf{L}_i (*left*), \mathbf{R}_i (*right*) y \mathbf{C}_i (*corner*) a las posiciones en el plano del fotograma i de los marcadores de la base de la cola, de las patas izquierda y derecha y de la esquina inferior izquierda del cilindro *rotarod*, respectivamente. A partir de estas, se calcularon las posiciones relativas $\tilde{\mathbf{C}}_i - \mathbf{T}_i$, $\mathbf{T}_i - \mathbf{L}_i$ y $\mathbf{T}_i - \mathbf{R}_i$. Es decir, que describiremos el estado del ratón observado en el fotograma i -ésimo con estos 3 vectores bidimensionales, los cuales representan posiciones relativas entre marcadores. Así, construimos un vector de estado \mathbf{x}_i , para cada fotograma i de video analizado, de la forma:

$$\mathbf{x}_i = \begin{bmatrix} \tilde{\mathbf{C}}_i - \mathbf{T}_i \\ \mathbf{T}_i - \mathbf{L}_i \\ \mathbf{T}_i - \mathbf{R}_i \end{bmatrix}, \quad (1.1)$$

donde $\tilde{\mathbf{C}}_i$ es la mediana de \mathbf{C}_i para cada prueba *rotarod* diferente (utilizamos su mediana pues la posición de la esquina del cilindro *rotarod* es constante en cada prueba pero varía entre días sucesivos). De esta manera el vector de estado \mathbf{x}_i tiene 6 componentes (pues está compuesto por 3 vectores bidimensionales) y existe una relación de correspondencia entre el estado en el fotograma i con su evolución temporal a tiempo t

$$\mathbf{x}_i \equiv \mathbf{x}(t). \quad (1.2)$$

Con el objetivo de reducir el ruido asociado a la detección exacta de cada marcador, las posiciones de los marcadores \mathbf{T}_i , \mathbf{L}_i , \mathbf{R}_i y \mathbf{C}_i se suavizaron utilizando un filtro de Saitzky–Golay [19] de ajuste polinomial de grado 3, con un ancho de ventana de 5 *frames* de

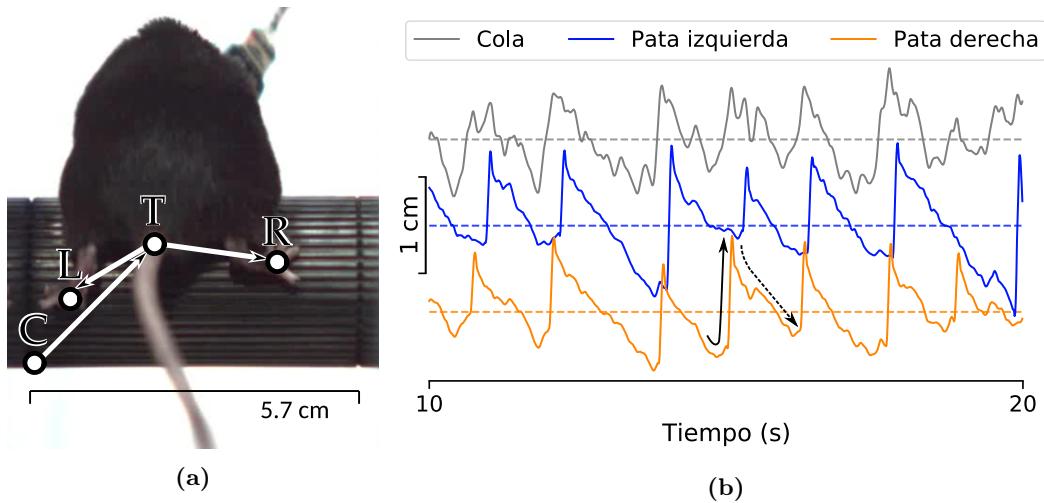


Figura 1.2: Caracterización de las posiciones de partes del cuerpo del ratón en función del tiempo. (a) Fotograma típico con las posiciones de los marcadores de la base de la cola y de las patas traseras (indicados por círculos). A partir de estas posiciones se calculan posiciones relativas (indicadas por flechas). (b) Serie temporal típica que muestra el movimiento sobre el eje vertical de los marcadores de la base de la cola y de las patas traseras (línea continua) y sus valores promedio en este intervalo de tiempo (línea discontinua). Con fines de visualización, los valores promedio de las series temporales se fijaron a 1 cm entre sí. Se marcó una instancia de movimiento activo (flecha en línea continua negra) y de movimiento pasivo (flecha en línea discontinua negra).

video. Para ello se utilizó la implementación en Python del filtro provista en el paquete de análisis de señales `scipy.signal`.

1.3. Algoritmo t-SNE

1.3.1. Introducción y formalismo

Habiendo extraído estos vectores de estado \mathbf{x}_i para cada fotograma i de video analizado podemos buscar alguna manera de visualizarlos que nos ayude a dilucidar su estructura subyacente. Para ello, utilizaremos el algoritmo t-SNE, la cual es una herramienta popular en la comunidad bioinformática, principalmente para la visualización de datos de dimensión alta. El objetivo principal de esta transformada es reducir la dimensión de un conjunto de datos preservando su estructura local.

La transformada t-SNE produce un mapeo uno a uno entre dos espacios: uno de dimensión alta y otro de dimensión baja (generalmente de 1 a 3 dimensiones). Este espacio de dimensión alta es el espacio de entrada (*input space*) o espacio de *features* de la transformada t-SNE, y denotaremos al conjunto de vectores pertenecientes a este espacio como $\{\mathbf{x}_i | i = 1, \dots, N\}$, siendo N es el número total de puntos. El espacio de dimensión baja, al que se quiere reducir nuestro conjunto de datos, es el espacio de salida (*output space*) de la transformada o espacio de *outcomes*. Denotaremos a este conjunto de vectores como $\{\mathbf{y}_i | i = 1, \dots, N\}$. Llamaremos a la imagen de la transformada t-SNE, o sea al conjunto de puntos $\{\mathbf{y}_i\}$, como mapa de comportamientos de los ratones o etograma. La Tabla 1.1 detalla los elementos básicos y la notación que se utilizará para la descripción de la transformada t-SNE.

La técnica t-SNE (*t-Distributed Stochastic Neighbor Embedding*), introducida por van

Espacio	Entrada (features)	Salida (outcomes)
Dimensión	Alta	Baja
Vector i -ésimo	\mathbf{x}_i	\mathbf{y}_i
Distancia	d , arbitraria	$\ \cdot\ $, usualmente L_2
Similaridad (probabilidades)	$p_{ij}(P)$	$q_{ij}(Q)$
Kernel	Gaussiano	t de Student

Tabla 1.1: Elementos básicos de la transformada t-SNE y notación utilizada. Comparación entre los espacios de entrada y de salida.

der Maaten y Hinton en 2008 [4], es la sucesora del algoritmo SNE (*Stochastic Neighbor Embedding*) el cual fue originalmente desarrollado por Hinton y Rosweis en 2002 [20]. La transformada t-SNE consiste en mapear puntos similares en el espacio de entrada en puntos cercanos en el espacio de salida, preservando en el proceso la estructura local del conjunto de datos. Para ello es necesario definir métricas en el espacio de entrada y en el de salida. Usualmente las implementaciones de t-SNE permiten definir una métrica generalizada, o una manera de medir distancia arbitraria, d en el espacio de entrada, mientras que convencionalmente se utiliza norma euclídea L_2 en el espacio de salida.

En nuestro caso, utilizamos en el espacio de entrada a la distancia coseno. Esta distancia se utiliza frecuentemente para analizar conjuntos de datos dispersos de dimensión alta en procesamiento de lenguaje natural (NLP, *Natural Language Processing*), es independiente de la magnitud de los vectores y mide las orientaciones relativas de dos vectores en el espacio [21].

1.3.2. Cálculo de similaridades

A partir de estas distancias, pueden definirse similaridades entre pares de puntos del espacio de entrada, utilizando un *kernel* gaussiano, de la siguiente manera

$$p_{j|i} = \frac{1}{Z_i} \exp \left\{ \left(-\frac{d(\mathbf{x}_i, \mathbf{x}_j)^2}{2\sigma_i^2} \right) \right\}, \quad (1.3)$$

donde la función de partición Z_i se calcula como

$$Z_i = \sum_{k \neq i} \exp \left\{ \left(-\frac{d(\mathbf{x}_i, \mathbf{x}_k)^2}{2\sigma_i^2} \right) \right\}. \quad (1.4)$$

De esta manera, la similaridad del punto \mathbf{x}_j , respecto del punto \mathbf{x}_i , es la probabilidad condicional $p_{j|i}$ de que el punto \mathbf{x}_i elija al punto \mathbf{x}_j como un vecino cercano. Esto sería si se eligieran vecinos cercanos en proporción a su densidad de probabilidad con una distribución gaussiana de distancias centradas en \mathbf{x}_i [4]. Fijamos $p_{i|i} = 0$, debido a que sólo nos interesa modelar similaridad entre pares de puntos diferentes. Estas $p_{j|i}$ también pueden interpretarse como la probabilidad de que ocurra una transición desde el estado \mathbf{x}_i en el fotograma i hacia un estado \mathbf{x}_j en el fotograma siguiente $i + 1$. Esto es así pues esperamos que para fotogramas consecutivos, el estado \mathbf{x} varíe continuamente, pues su movimiento también es continuo. La introducción de estas probabilidades $p_{j|i}$ es lo que le da el nombre de “estocástico” (*stochastic*) al algoritmo t-SNE y el hecho de que estas probabilidades sean de transición entre vecinos cercanos es también lo que le da la parte de “neighbor” a su nombre.

Por último, estas similaridades se simetrizan de manera que sean equivalentes sin importar el orden en que se tomen los puntos del par $(\mathbf{x}_i, \mathbf{x}_j)$

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}, \quad (1.5)$$

donde N es el número total de puntos en el espacio de entrada. Este paso de simetrización fue introducido por primera vez en una variante de SNE conocido como SSNE (*Symmetric Stochastic Neighbor Embedding*) [22]. Note además que $p_{ii} = 0$ y que $\sum_{i,j} p_{ij} = 1$, por lo que p_{ij} resulta una distribución de probabilidad válida sobre el conjunto de todos los pares posibles de puntos $(\mathbf{x}_i, \mathbf{x}_j)$ [9]. Simetrizar las probabilidades hace que no nos importe el orden de los fotogramas en los que estudiamos las transiciones.

La desviación estándar σ_i del *kernel* gaussiano se calcula para cada punto \mathbf{x}_i del espacio de entrada de manera de mantener constante un parámetro predefinido \mathcal{P} , generalmente fijado en $\mathcal{P} = 30$, denominado perplejidad (*perplexity*)

$$\mathcal{P}_i = \exp\{H_i\} \quad (1.6)$$

$$= \mathcal{P} \text{ (constante),} \quad (1.7)$$

siendo H_i la entropía de Shannon de la distribución de probabilidad condicional $p_{j|i}$ de *kernel* gaussiano definida anteriormente

$$H_i = - \sum_j p_{j|i} \ln p_{j|i}. \quad (1.8)$$

El parámetro de perplejidad \mathcal{P} está relacionado con qD_i , el índice de diversidad generalizado de orden q (*generalized diversity index*) [23, 24], definido como

$${}^qD_i = \left(\sum_j p_{j|i} \right)^{1/(1-q)}, \quad (1.9)$$

el cual, en el límite cuando $q \rightarrow 1$ equivale a la perplejidad

$${}^1D_i = \lim_{q \rightarrow 1} {}^qD_i \quad (1.10)$$

$$= \mathcal{P}_i. \quad (1.11)$$

Podemos pensar que la perplejidad es el número de vecinos más cercanos a los que transiciona el estado \mathbf{x}_i del ratón. Aumentar (o reducir) este parámetro aumentará (o reducirá) entonces la cantidad total de vecinos que serán más relevantes para nuestras transiciones entre fotogramas. Además, usar una perplejidad demasiado baja nos daría una descripción miope de las transiciones posibles desde el estado \mathbf{x}_i del ratón (podría sesgar nuestro mapa t-SNE), mientras que usar una perplejidad demasiado alta incluiría demasiados vecinos como cercanos. Esto último podría aumentar la variabilidad del mapa t-SNE, pero más grave para nuestros fines prácticos es que aumentaría mucho la complejidad de cómputo de del algoritmo t-SNE.

El número qD_i se conoce también como número efectivo de especies, y está definido en este caso en torno al punto \mathbf{x}_i . El número qD_i es una medida de la cantidad de puntos \mathbf{x}_j (con $j \neq i$) que son similares a \mathbf{x}_i . Esto es, está relacionado con el concepto de vecinos cercanos al estado \mathbf{x}_i .

Existe un motivo por el cual la desviación estándar σ_i se selecciona de manera tal que la perplejidad \mathcal{P}_i se mantenga constante para cada punto \mathbf{x}_i del espacio de entrada. Pues así se permite que el ancho del *kernel* gaussiano σ_i se adapte a las condiciones locales del conjunto de datos. Esto es, en regiones de densidad alta (baja) del conjunto de datos se utilizan valores más pequeños (grandes) de σ_i . Note además que la entropía H_i aumenta al aumentar σ_i [4]. Esto es en parte lo que asegura que el mapa t-SNE preserve la estructura local de nuestro conjunto de datos.

Por otro lado, y de manera similar a como hicimos en el espacio de entrada, se definen similaridades en el espacio de salida de la forma

$$q_{ij} = \frac{1}{Z} \frac{1}{1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2}, \quad (1.12)$$

donde la función de partición \mathcal{Z} se calcula como

$$\mathcal{Z} = \sum_k \sum_{l \neq k} \frac{1}{1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2}. \quad (1.13)$$

De esta manera, las similaridades en el espacio de salida se calculan utilizando un *kernel* de Cauchy (t de Student con grado de libertad $\nu = 1$). El objetivo del mapa t-SNE es que estas probabilidades q_{ij} reflejan de la manera más fiel posible a las probabilidades p_{ij} de nuestros datos. Esta es la clave para que el algoritmo t-SNE sea capaz de brindar una representación en un espacio, de por ejemplo 2 dimensiones, de un conjunto de datos de dimensión mayor, por ejemplo en nuestro caso de 6 dimensiones. Y esto es a lo que nos referimos cuando decimos que t-SNE preserva la estructura local de nuestros datos.

Ahora, un poco más de historia. Originalmente, el algoritmo SNE utilizaba un *kernel* gaussiano para calcular las similaridades en el espacio de salida. Esta era la principal causa del llamado problema de hacinamiento (*crowding problem*) [4, 20]. Este problema consistía en una aglomeración de puntos en el centro del mapa, dificultando la visualización de diferentes *clusters* de datos. Una de las primeras instancias para solucionar el problema fue la implementación de un *kernel* con colas pesadas uniformes, técnica que se denominó UNI-SNE (haciendo alusión a la incorporación de una distribución uniforme), la cual añadía un término constante en el cálculo de la distribución de probabilidad q_{ij} [25]. Finalmente, prevaleció la implementación t-SNE que utilizamos actualmente, la cual utiliza un *kernel* de distribución t de Student en el espacio de salida, y esta es una distribución de colas pesadas.

La implementación de t-SNE puede generalizarse utilizando *kernels* t de Student con grados de libertad ν arbitrarios [26]. Esto nos permite obtener mapas t-SNE con *clusters* mejor definidos. De esta manera, si se define un nuevo parámetro α tal que $\nu = 2\alpha - 1$, el *kernel* adopta la forma

$$q_{ij} = \frac{1}{\mathcal{Z}} \frac{1}{(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2/\alpha)^\alpha}, \quad (1.14)$$

donde esta vez la función de partición \mathcal{Z} se calcula como

$$\mathcal{Z} = \sum_k \sum_{l \neq k} \frac{1}{(1 + \frac{1}{\alpha} \|\mathbf{y}_k - \mathbf{y}_l\|^2/\alpha)^\alpha}. \quad (1.15)$$

Por último, fijamos $q_{ii} = 0$. Note que $\sum_{i,j} q_{ij} = 1$ por lo que q_{ij} resulta, al igual que p_{ij} , una distribución de probabilidad válida sobre el conjunto de todos los pares de puntos, esta vez $(\mathbf{y}_i, \mathbf{y}_j)$. Definamos entonces como P y Q a los conjuntos de probabilidades $\{p_{ij}\}$ y $\{q_{ij}\}$ sobre todos los pares posibles (i, j) , respectivamente.

Ahora que hemos calculado estas probabilidades de transición P y Q , ¿cómo hacemos para obligar a las probabilidades Q a parecerse más a las probabilidades P ? Esto se hace definiendo una función de costo, que penalice las diferencias entre P y Q . De esta manera, minimizar la función de costo sería equivalente a obtener un mapa t-SNE en donde las probabilidades Q mejor reflejan a las probabilidades P .

1.3.3. Función de costo

Como dijimos, el objetivo del algoritmo t-SNE es conseguir una representación de baja dimensión ($\{\mathbf{y}_i\}$) de nuestros datos ($\{\mathbf{x}_i\}$) que mejor conserve las similaridades entre pares de puntos. De esta manera, podemos capturar en el análisis el concepto de que el animal no se mueve instantáneamente de un extremo a otro del espacio, sino que hay una relación de proximidad entre puntos. Para este fin se utiliza la divergencia de Kullback-Leibler (KLD, *Kullback-Leibler Divergence*) como función de costo. La KLD es una medida de qué tan parecidas son dos distribuciones de probabilidad y en nuestro caso coincide con la entropía

cruzada de las distribuciones P y Q a menos de una constante [4]. De esta manera, el proceso de optimización de un mapeo t-SNE consiste en minimizar la siguiente función de costo

$$\mathcal{L} = D_{\text{KL}}(P||Q) \quad (1.16)$$

$$= \sum_{i \neq j} p_{ij} \log_2 \frac{p_{ij}}{q_{ij}}. \quad (1.17)$$

Por un lado, si las probabilidades P y Q son exactamente iguales, entonces la función de costo es cero (este es el mínimo valor que puede adquirir), por otro lado, si las probabilidades P y Q son muy diferentes, la función de costo puede adoptar un valor arbitrariamente grande. Además esta función de costo penaliza más fuertemente cuando fallamos en modelar probabilidades de transición p_{ij} altas, que cuando fallamos en modelar p_{ij} bajas. Esto permite que nuestro mapa t-SNE refleje mejor las características más salientes de nuestros datos.

La minimización de la función de costo se realiza mediante descenso por gradientes (*gradient descent*). Dentro de la expresión del gradiente pueden identificarse dos términos

$$\frac{1}{4} \frac{\partial \mathcal{L}}{\partial \mathbf{y}_i} = \varepsilon \underbrace{\sum_{j \neq i} p_{ij} w_{ij}^{1/\alpha} (\mathbf{y}_i - \mathbf{y}_j)}_{\text{Término atractivo}} - \underbrace{\sum_{j \neq i} q_{ij} w_{ij}^{1/\alpha} (\mathbf{y}_i - \mathbf{y}_j)}_{\text{Término repulsivo}}, \quad (1.18)$$

donde se agrega al término atractivo un factor multiplicativo ε denominado exageración (*exaggeration*), cuyo valor por defecto es $\varepsilon = 1$ [27] y definiendo además a w_{ij} en la expresión (1.18) como

$$w_{ij} = (1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2 / \alpha)^{-\alpha}. \quad (1.19)$$

Una manera de interpretar el gradiente de la función de costo es como si fueran sumas de fuerzas de “resortes” que unen a todos los pares de puntos $(\mathbf{y}_i, \mathbf{y}_j)$ del mapa t-SNE. Así, un punto \mathbf{y}_i del mapa t-SNE sufre fuerzas que pueden ser tanto atractivas como repulsivas en las direcciones $\mathbf{y}_i - \mathbf{y}_j$. Siguiendo esta interpretación, optimizar el mapa t-SNE, que equivale a minimizar la función de costo, equivale también a encontrar el equilibrio de las fuerzas de todos estos resortes [25].

Finalmente, las posiciones \mathbf{y}_i del mapeo t-SNE se actualizan, al final de una determinada iteración n , de acuerdo a la siguiente regla [4, 22]

$$\mathbf{y}_i^{(n+1)} = \mathbf{y}_i^{(n)} + \eta \frac{\partial \mathcal{L}}{\partial \mathbf{y}_i} \Big|_{\mathbf{y}_i=\mathbf{y}_i^{(n)}} + \beta^{(n)} \left(\mathbf{y}_i^{(n)} - \mathbf{y}_i^{(n-1)} \right), \quad (1.20)$$

siendo η la tasa de aprendizaje (*learning rate*) y $\beta^{(n)}$ el *momentum* en la iteración n .

Generalmente, las posiciones \mathbf{y}_i del mapeo t-SNE se inicializan como las primeras componentes principales (PCA, *Principal Component Analysis*) del conjunto de datos \mathbf{x}_i . Luego, el proceso de optimización se repite usualmente alrededor de $n_{\text{iter}} = 1000$ iteraciones. Originalmente, en la implementación SNE de Hinton, las posiciones \mathbf{y}_i se inicializaban de manera aleatoria [20]. Utilizar PCA para inicializar el mapa t-SNE tiene la ventaja de obtener mapas que son reproducibles, pues se elimina la aleatoriedad en su inicialización. Además, inicializar con PCA mejora la estructura global del mapa t-SNE, pues se basa en representar nuestro conjunto de datos a lo largo de las direcciones de mayor varianza del mismo, brindando una primera descripción “a grandes rasgos” [9].

Para adentrarse más en los detalles de implementación y para saber más sobre diferentes técnicas utilizadas en la optimización de los mapas t-SNE, por favor referirse a los Apéndices A.1 y A.2.

1.3.4. Obtención del mapa de comportamientos

En este trabajo se utilizó la implementación en Python `openTSNE` [28] del algoritmo t-SNE. También existen implementaciones de paramétricas de t-SNE en `TensorFlow` y `Keras`, aunque la primera implementación de este tipo fue desarrollada en MatLab [4]. La implementación *benchmark* del algoritmo en Python es la de `scikit-learn` [29]. La implementación `openTSNE` nos brinda una gran flexibilidad en el control de los parámetros y del proceso de optimización. El Apéndice A.3 brinda un resumen de los parámetros utilizados en nuestro mapa.

La Figura 1.3 muestra el mapa comportamientos o etograma obtenido al aplicar la transformada t-SNE sobre el conjunto de vectores de estado $\{\mathbf{x}_i\}$. Este mapa se construyó usando todos los fotogramas analizados correspondientes a los $N_{\text{ratones}} = 3$ ratones observados. Recordemos que el vector de estado \mathbf{x}_i contiene información sobre las posiciones relativas de algunas partes del cuerpo del ratón en el fotograma i -ésimo, durante la ejecución de la tarea motora *rotarod* con aceleración. Por lo tanto, este mapa es una representación de las posiciones relativas de los ratones y de sus partes del cuerpo.

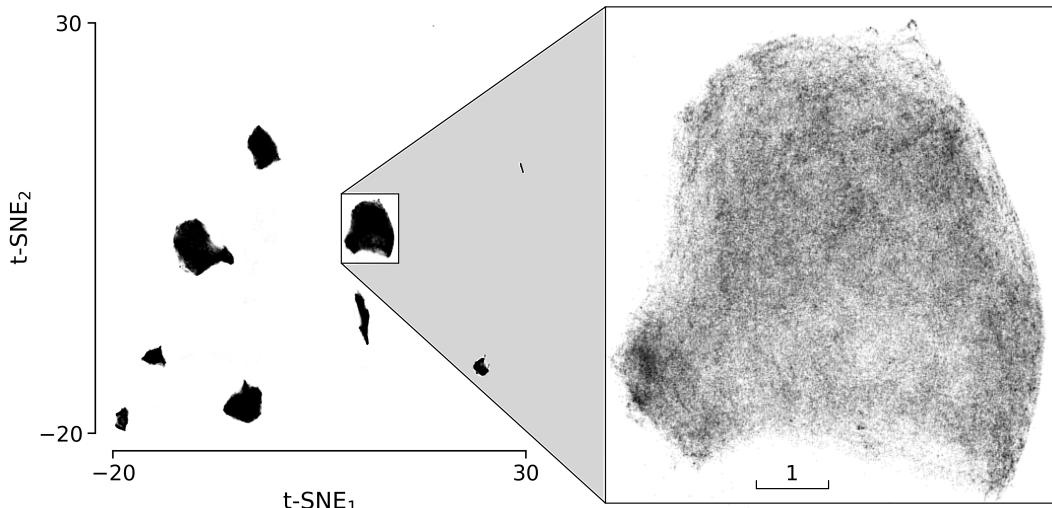


Figura 1.3: Etograma obtenido a partir de la transformada t-SNE del conjunto de vectores de estado $\{\mathbf{x}_i\}$ de todos los *frames* de video analizado. Estos vectores de estado representan las posiciones relativas de algunas partes del cuerpo de los ratones. Dentro del recuadro se muestra más en detalle los puntos de una región del etograma.

En el etograma de la Figura 1.3 puede observarse la aparición de *clusters* de puntos. Cada punto \mathbf{y}_i en el mapa t-SNE de la Figura 1.3 es la transformada de un vector de estado \mathbf{x}_i . Además, sabemos que la transformada t-SNE conserva la estructura local del conjunto de datos $\{\mathbf{x}_i\}$ y que puntos \mathbf{y}_i cercanos en el mapa t-SNE se corresponden con vectores \mathbf{x}_i similares. Entonces podríamos pensar que cada uno de los *clusters* que se observan en el mapa t-SNE representan subconjuntos de vectores de estado $\{\mathbf{x}_i\}_l$ similares, donde l es el número de clase o de pertenencia al *cluster* l -ésimo. Llamaremos *labels* a estos números de clase o de pertenencia al *cluster* l -ésimo. En la siguiente sección detallaremos el proceso de segmentación del etograma, mediante el cual asignaremos a cada punto del mapa t-SNE un *label* determinado. Para ello utilizaremos la transformación divisoria (*watershed*).

1.4. Transformación *watershed*

Si bien el mapa t-SNE obtenido presenta *clusters* muy bien definidos, quisimos segmentar estos *clusters* de manera automática e independiente del usuario. Para ello utilizamos la transformación *watershed*, que es un algoritmo de segmentación de imágenes en escala de grises. Una imagen en escala de grises puede pensarse como un relieve topográfico, donde el nivel de gris de cada píxel se interpreta como su altura en el mapa de relieve. Cuando se vuelca agua sobre un relieve topográfico, esta fluye a lo largo de diferentes caminos hasta finalmente llegar a un mínimo local de potencial. Intuitivamente, la divisoria de un relieve corresponde a los límites de las cuencas hidrográficas adyacentes. Esencialmente esta es la interpretación del funcionamiento de la transformación *watershed* [30, 31].

La transformación *watershed* presenta una gran flexibilidad en cuanto al número total de regiones que se pueden segmentar en el plano, ya que dicho valor no requiere estar pre-determinado. Además no representa una limitación en cuanto a complejidad computacional, ya que en nuestro caso se aplica directamente sobre la densidad de probabilidad de nuestro mapa t-SNE, y no sobre el conjunto de todos los puntos $\{y_i\}$ del mapa.

La Figura 1.4a muestra una estimación de densidad de probabilidad con *kernel* gaussiano (KDE) de ancho de banda (*bandwidth*) $\sigma = 1$, para la distribución de puntos en el mapa t-SNE de la Figura 1.3. La densidad de probabilidad se calculó utilizando el módulo `fastKDE` [32] implementado en Python, junto con un filtro de convolución gaussiano de la librería `scikit-image` [33]. La implementación `fastKDE` aprovecha la transformada rápida de Fourier no uniforme (nuFFT), para conseguir una complejidad computacional $O(N)$, lo cual es muy eficiente para nuestros propósitos. La densidad de probabilidad se calculó sobre una cuadrícula con una resolución de 2049×2049 px². Esta resolución es más que suficiente para nuestra tarea de segmentación.

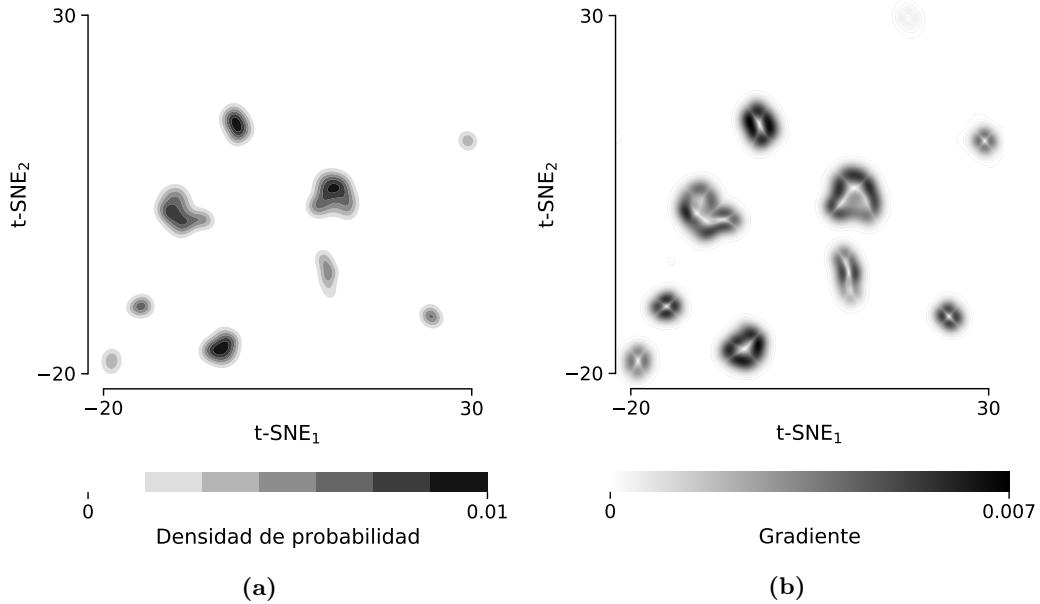


Figura 1.4: Estimación de la distribución de densidad de probabilidad con *kernel* gaussiano (KDE) de puntos del mapa t-SNE de la Figura 1.3. (a) Superficies de nivel de la KDE. (b) Gradiente de la KDE.

La densidad de probabilidad que se muestra en la Figura 1.4a presenta picos rodeados por amplios valles. En nuestro caso, nos interesa segmentar el mapa t-SNE en torno a estos

picos de alta probabilidad y no en torno a las cuencas de baja probabilidad, pues lo que queremos es encerrar a los picos dentro de una única región. Por este motivo inicializamos la transformación *watershed* utilizando un conjunto de máximos locales de la densidad de probabilidad. Por su parte, la Figura 1.4b muestra el gradiente de la densidad de probabilidad de nuestro mapa t-SNE. La transformación *watershed* encuentra regiones conexas en el mapa t-SNE, tales que si subimos a favor del gradiente de densidad de probabilidad dentro de una de estas regiones, llegamos a un mismo máximo local.

Cabe mencionar que antes de realizar la transformación *watershed* se aplica un filtro de mediana usando una máscara de disco de radio de 2 px para reducir el ruido de la imagen de la densidad de probabilidad, y el resultado es el que se muestra en la Figura 1.4a. Luego, se aplica un filtro de gradiente a la imagen, obteniéndose así la Figura 1.4b. Este filtro de gradiente se aplicó también con una máscara de disco de 2 px de radio. De esta manera se obtiene la imagen que funcionará como guía para delimitar las fronteras entre las distintas regiones en la segmentación del mapa t-SNE.

Finalmente la Figura 1.5 muestra el mapa de comportamientos segmentado en $N_{\text{clases}} = 9$ regiones diferentes, y las fronteras entre regiones contiguas. Cada una de estas regiones encierra un *cluster* bien definido e incluye algunos puntos fuera del *cluster* principal a los cuales se les asigna una misma etiqueta de pertenencia de clase, es decir un *label* único. Sobre la Figura 1.5 se anotaron los números de *label* asignados a cada región segmentada. Los números de *label* se colocaron en la posición del centro de masa de cada región. Note que la región de *label* número 8 es la única que encierra dos *clusters* separados, aunque el centro de masa de esta región está ubicado encima de uno de ellos, por lo que el peso del *cluster* más alejado es despreciable en comparación.

En el siguiente capítulo estudiaremos algunas propiedades del etograma obtenido a través del algoritmo t-SNE. También exploraremos e interpretaremos los *labels* que resultaron de la segmentación mediante transformación *watershed*.

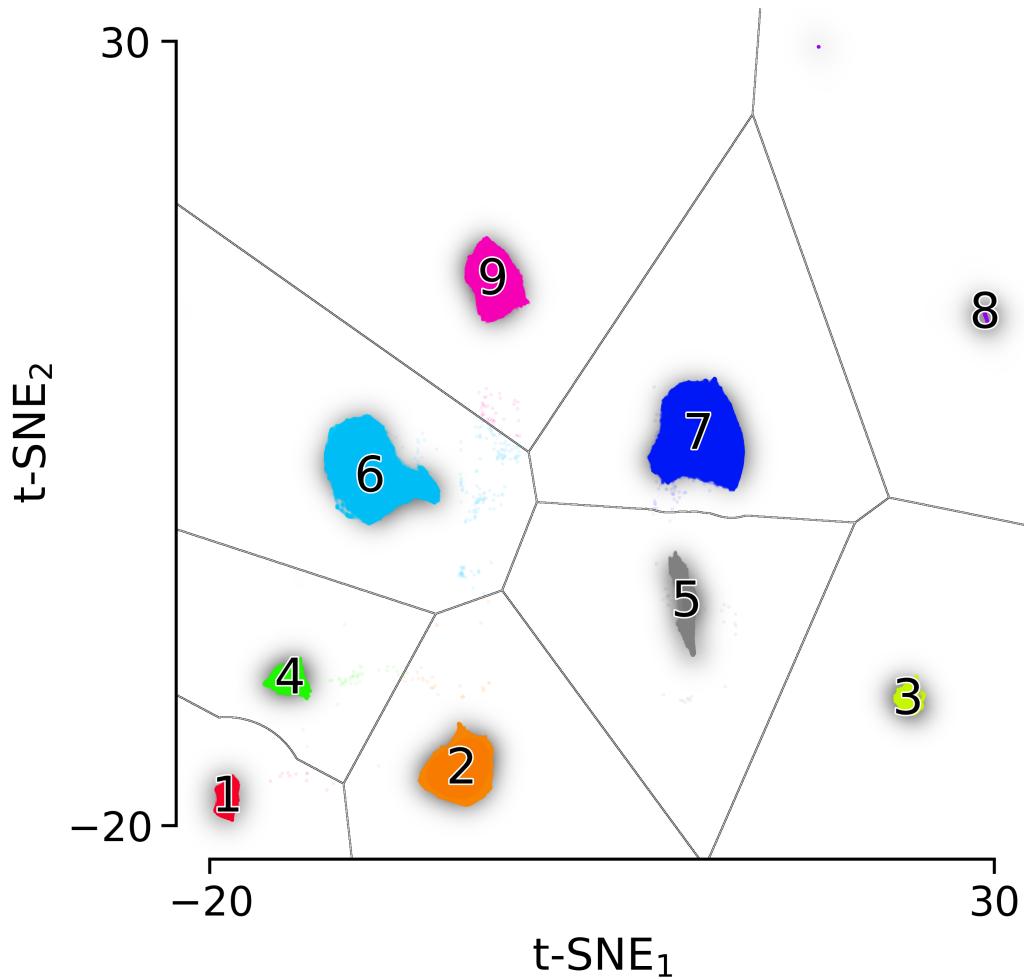


Figura 1.5: Mapa de comportamientos obtenido a partir del conjunto de vectores de estado $\{\mathbf{x}_i\}$ de todos los fotogramas de video analizados. En este mapa t-SNE se anotaron los números de clase (*labels*) asignados a cada región segmentada. Estas anotaciones se posicionaron encima de los centros de masa de cada región. Se marcaron las fronteras entre regiones contiguas. Los puntos están coloreados de acuerdo al *label*.

Capítulo 2

Análisis e interpretación del mapa de comportamientos

Una vez obtenida la segmentación del mapa de t-SNE, nos propusimos analizar si los *labels* obtenidos se corresponden con determinados comportamientos o poses del animal observables por el experimentador, es decir, si podemos interpretar el mapa obtenido. Dado que t-SNE es un algoritmo de clasificación no supervisada, es posible que algunos o todos los *labels* obtenidos no sean interpretables. En este capítulo estudiaremos las propiedades del mapa de comportamientos obtenidos en el Capítulo 1. También analizaremos las propiedades de los *labels* obtenidos a partir de la segmentación del etograma y los interpretaremos como poses específicas adoptadas por los ratones expertos durante la ejecución de la tarea *rotarod* con aceleración.

A continuación damos un resumen del contenido de este capítulo. Primero, en la Sección 2.1 daremos significado a los *labels* que surgieron de la segmentación del etograma. Luego, en la Sección 2.2 estimaremos la representatividad de cada clase de *labels* dentro del conjunto de ratones individuales. Después, en la Sección 2.3 caracterizaremos la estadística de las secuencias de *labels* y su dependencia con la velocidad del cilindro *rotarod*. Finalmente, en la Sección 2.4 representaremos las secuencias de *labels* como logos de secuencias condicionadas a la ocurrencia de una determinada clase de *label*.

2.1. Interpretación del mapa

Como primera medida nos interesa interpretar los *labels* que surgieron de la segmentación del mapa de comportamientos de la Figura 1.5. Esto es, interpretar la clasificación no supervisada del comportamiento animal producida utilizando el algoritmo t-SNE. Para ello se compilaron los intervalos de video donde ocurría un *label* determinado por ratón y por *trial*, de manera de poder visualizar la similitud en el comportamiento asociado a cada *label* ([link al material suplementario](#)).

Además, la Figura 2.1 muestra ejemplos representativos de los fotogramas extraídos. Notablemente, logramos interpretar los *labels* de la segmentación del etograma como diferentes tipos de poses que los ratones adoptan durante la ejecución de la tarea. Por ejemplo, el *label* 2 (7) estaría asociado a una pose asimétrica en la que el ratón da un paso con su pata derecha (izquierda). Por otro lado, los *labels* 5, 6 y 9 estarían asociados a poses simétricas y podrían ser diferentes poses intermedias entre dos pasos consecutivos del ratón o momentos en los que el ratón avanza o salta de manera sincronizada con ambas patas. Por su parte, el *label* 1 (8) podría estar asociado a poses en las que el ratón oculta su pata derecha (izquierda). Por último, las poses asociadas a los *labels* 3 y 4 no fueron interpretadas. En conclusión,

la clasificación del comportamiento obtenida nos permite identificar poses individuales del animal de corta duración mientras realiza la tarea.

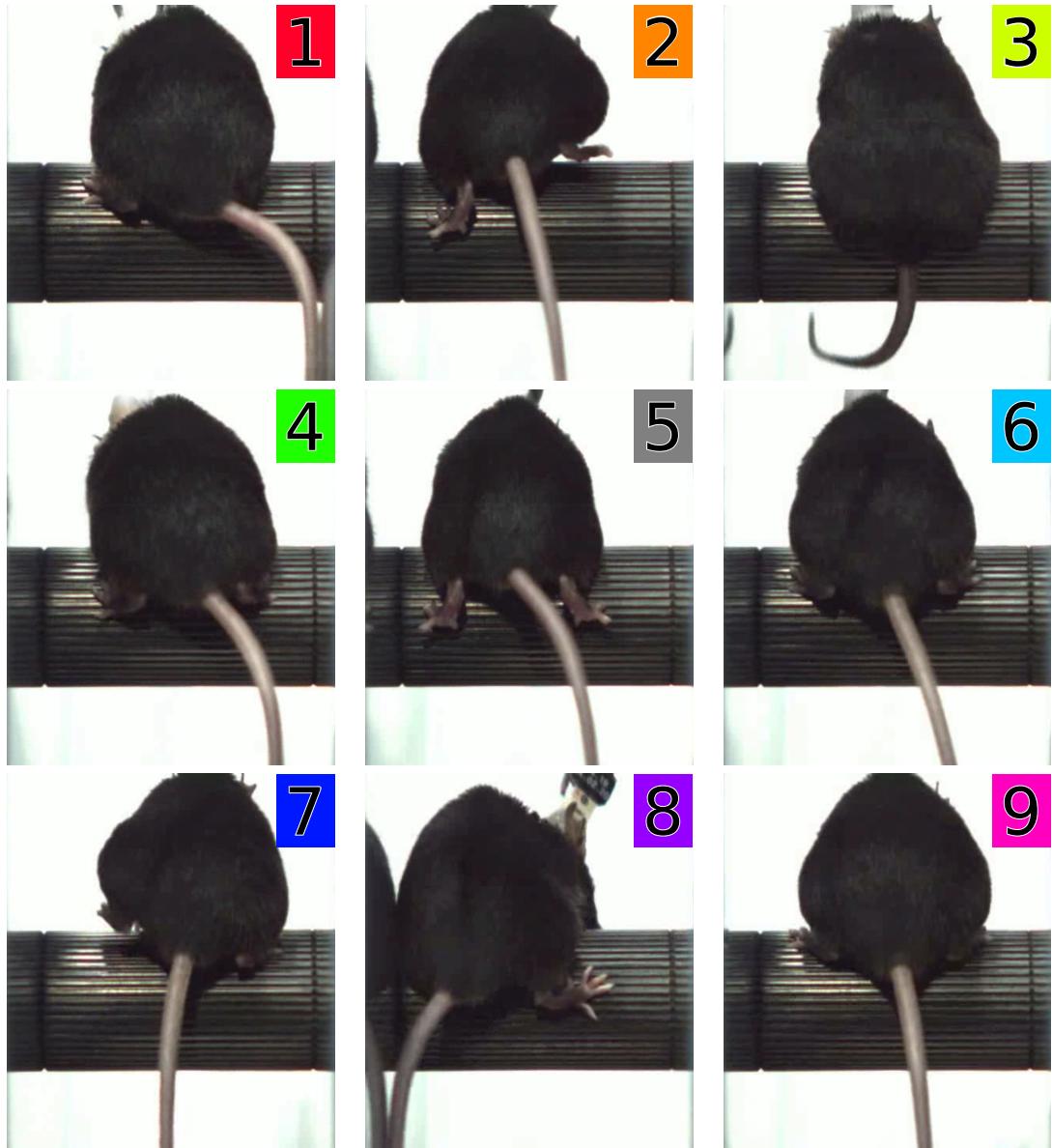


Figura 2.1: Fotogramas típicos extraídos de los videos de los ratones expertos durante la realización de la tarea *rotarod*. Se extrajo un fotograma representativo por cada clase de *label*, obtenido a partir de la segmentación del mapa de comportamientos de la Figura 1.5.

Por su parte, la Figura 2.2 muestra una sucesión típica de *labels*, durante un intervalo de tiempo de 10 s, comenzando a los 10 s de una prueba *rotarod*. En la parte superior de la Figura 2.2 se muestran además las coordenadas verticales de las patas izquierda y derecha del ratón. En la figura se marcaron con círculos los máximos locales alcanzados por las posiciones de las patas. Puede observarse que la ocurrencia del *label* 7 (2) está asociada con el movimiento de la pata izquierda (derecha), apoyando nuestra interpretación de que el *label* 7 (2) está asociado a una pose en la que el ratón da un paso con su pata izquierda

(derecha).

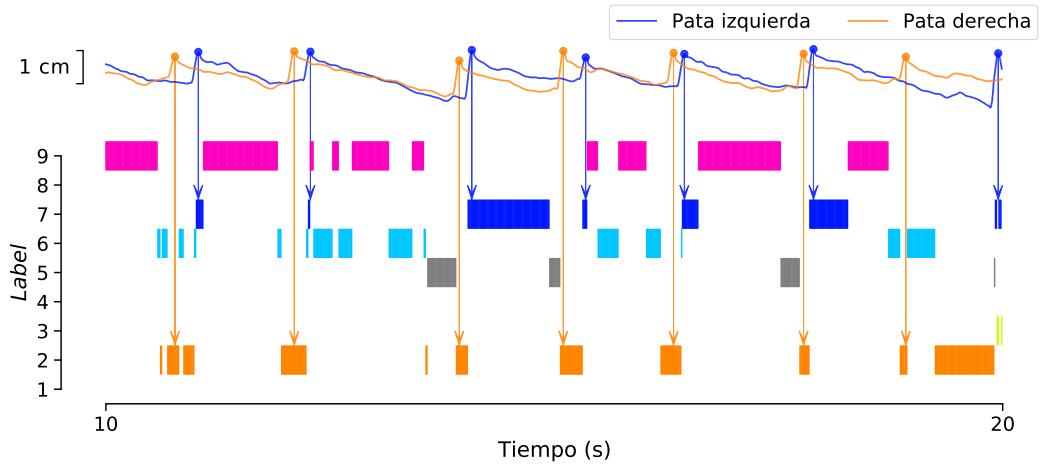


Figura 2.2: Serie temporal típica de *labels*. En la parte superior se muestran la evolución temporal de las coordenadas verticales de las patas izquierda y derecha del ratón. Se indican con círculos los máximos de las posiciones de las patas izquierda y derecha y se trazan flechas hacia los correspondientes *labels* 7 y 2.

2.2. Representatividad de la clasificación

Una vez que logramos interpretar los *labels* obtenidos, nos preguntamos si la existencia de diferentes poses estaba relacionada a diferentes estrategias comportamentales en animales particulares o si bien eran parte de un repertorio común de poses. Para ello nos preguntamos qué tan representativos son las regiones segmentadas del etograma de la Figura 1.5. Más precisamente, queremos averiguar si todos los ratones visitan cada una de las regiones del mapa en proporciones similares. La Figura 2.3 muestra los puntos correspondientes a cada ratón por separado en el etograma (recordemos que en esta práctica se trabajó con un conjunto de $N_{\text{ratones}} = 3$ ratones distintos) donde se observa que cada *label* es visitado efectivamente por los tres ratones.

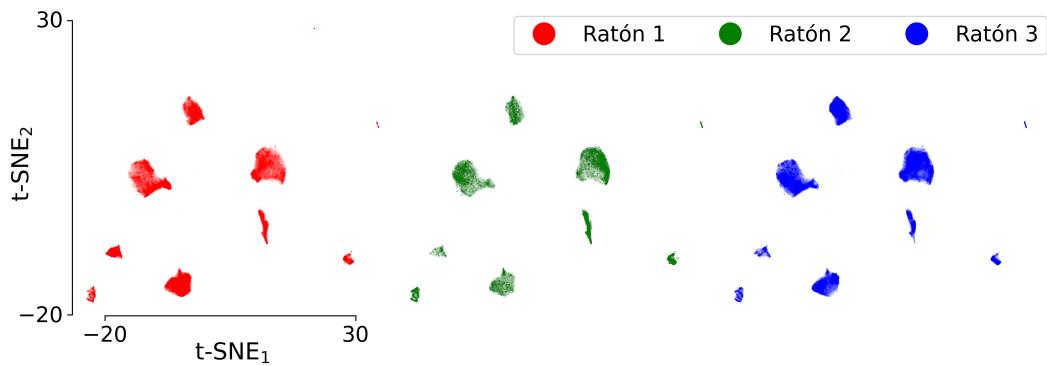


Figura 2.3: Puntos correspondientes a cada ratón por separado en el etograma obtenido.

Con el fin de cuantificar la representatividad de las muestras de ratones de cada región segmentada del etograma calcularemos los índices de diversidad generalizados de orden 2

2D_l para cada clase de *label* l . Este índice de diversidad 2D_l , también conocido como número efectivo de individuos, es una medida de cuántos ratones diferentes visitan una determinada clase de *label* l . Para ello, primero debemos calcular las probabilidades de coincidencia de ratón λ_l condicionales a cada clase de *label* l . Para una dada clase de *label* $l \in \{1, \dots, N_{\text{clases}}\}$ con $N_{\text{clases}} = 9$, se tiene que la probabilidad de que 2 muestras sean del mismo ratón, dado que comparten una misma clase de *label* es

$$\lambda_l = \sum_{r=1}^{N_{\text{ratones}}} p_{r|l}^2, \quad (2.1)$$

siendo $N_{\text{ratones}} = 3$ el número total de ratones distintos observados y $p_{r|l}$ la probabilidad condicional

$$p_{r|l} = P(\text{ratón sea } r \mid \text{label es de clase } l). \quad (2.2)$$

Dado que la cantidad de observaciones totales para cada ratón no es la misma, vamos a calcular $p_{r|l}$ normalizando la ocurrencia de un *label* de clase l para cada ratón r . Para eso, vamos a calcular las probabilidades condicionales

$$p_{l|r} = P(\text{label sea de clase } l \mid \text{ratón es } r) \quad (2.3)$$

$$= \frac{n_{l|r}}{N_r}, \quad (2.4)$$

donde $n_{l|r}$ es el número de observaciones del *label* clase l para el ratón r y N_r es el número total de observaciones del ratón r . Note que $\sum_{l=1}^{N_{\text{clases}}} p_{l|r} = 1 \forall r$.

La Tabla 2.1 muestra los valores de las probabilidades condicionales obtenidas $p_{l|r}$. Estos valores muestran que hay variabilidad en la probabilidad de ocurrencia de poses, siendo más frecuentes los *labels* 2, 6, 7 y 9, y el *label* 8 el menos frecuente. También podemos observar la diferencia inter-animal en la probabilidad de ocurrencia de cada *label*.

A partir de las $p_{l|r}$ podemos calcular las $p_{r|l}$ independientemente de la cantidad total de observaciones de cada ratón, de la siguiente manera

$$p_{r|l} = p_{l|r} \frac{1}{\sum_{r=1}^{N_{\text{ratones}}} p_{l|r}}. \quad (2.5)$$

Note que estas probabilidades condicionales satisfacen $\sum_{r=1}^{N_{\text{ratones}}} p_{r|l} = 1 \forall l$.

Por un lado, si una dada clase de *label* l estuviera ocupada exclusivamente por un sólo tipo de ratón, entonces la probabilidad de coincidencia sería $\lambda_l = 1$. En cambio, si un tipo de *label* l fuera visitado por todos los ratones con probabilidades iguales, entonces la probabilidad de coincidencia de ratones sería $\lambda_l = 1/N_{\text{ratones}} = 1/3$. Además, las probabilidades de coincidencia satisfacen que

$$\frac{1}{N_{\text{ratones}}} \leq \lambda_l \leq 1 \forall l. \quad (2.6)$$

Esta probabilidad de coincidencia supone que se toman muestras al azar con repetición. Este supuesto puede fallar en caso de contar con pocas observaciones de ratones en cada *label*. En nuestro caso, se cuentan con observaciones totales del orden de 10^5 para cada ratón y de órdenes entre 10^3 y 10^4 para cada tipo de *label* visitado por cada ratón.

Finalmente, a partir de las probabilidades de coincidencia λ_l podemos calcular el número efectivo de individuos presentes en cada *label* como

$${}^2D_l = \frac{1}{\lambda_l}. \quad (2.7)$$

Luego, los valores extremos que puede adoptar el índice de diversidad 2D_l corresponden a ${}^2D_l = 1$ para el caso de una clase de *label* l ocupada por un único tipo de ratón y

${}^2D_l = N_{\text{ratones}} = 3$ para el caso de ocupación equiprobable. En la Tabla 2.1 se indican los valores de los índices de diversidad calculados a partir de esas probabilidades condicionales. Para cada *label* l se resaltaron los valores mayores ${}^2D_l = 2.5$. Note que los valores obtenidos satisfacen que $1 \leq {}^2D_l \leq 3 \forall l$. A partir de este análisis observamos que los *labels* más frecuentes (2, 6, 7 y 9) son visitados similarmente por los tres ratones, pero que también hay *labels* de baja probabilidad y alta representatividad (por ejemplo los *labels* 3 y 5). Por el contrario, el *label* 4 corresponde a una pose que representa con mayor probabilidad al ratón 1.

<i>label</i>	1	2	3	4	5	6	7	8	9
Ratón 1	3	23	4	11	7	22	20	1	11
$p_{l r} (\%)$	6	11	6	2	10	23	30	2	10
Ratón 3	2	13	4	2	6	25	28	2	18
Diversidad 2D_l	2.46	2.70	2.83	1.82	2.83	2.99	2.92	2.43	2.78

Tabla 2.1: Frecuencia condicional $p_{l|r}$ de observar el *label* l del etograma de la Fig. 1.5 con el r -ésimo ratón. Valores de diversidad 2D_l calculados para cada *label* l . Se resaltan en negrita aquellos valores de diversidad mayores a ${}^2D_l = 2.5$.

2.3. Relación entre el comportamiento y la velocidad del rotarod

Las series temporales de *labels* de la Figura 2.2 evidencian que los *labels* tienen diferentes duraciones o tiempos de permanencia. Dado que el paradigma de rotarod utilizado incluye una aceleración positiva constante, nos preguntamos cómo depende la duración de las poses con la velocidad del rotarod y encontramos que las duraciones promedio de los *labels* disminuyen con la velocidad del *rotarod* (Figura 2.4a). Este resultado está de acuerdo a lo esperado de manera intuitiva: a mayor velocidad de *rotarod*, más rápido cambia de pose el ratón. Para velocidades medias a altas los *labels* 2, 6, 7 y 9 son los de mayor duración (en torno a los 100 ms), mientras que el resto tienen duraciones de entre 30 y 50 ms. Sin embargo, es importante notar que la reducción en la duración de ciertos *labels* no es proporcional con el cambio en velocidad (*labels* 4, 6 y 9) sino que se reduce de manera abrupta indicando un posible cambio en la estrategia comportamental elegida.

Luego, nos preguntamos si es posible distinguir cambios en la estrategia comportamental asociados a diferentes velocidades del rotarod. Para ello, calculamos la probabilidad de ocurrencia de cada *label* a velocidad baja, media o alta independientemente de la duración del mismo. Dado que los *labels* tienen duraciones variables, el estudio de su probabilidad incluyendo la duración podría traer como consecuencia la sobre-representación de los *labels* con mayor duración. En cambio, de la manera propuesta, intentamos poner a los *labels* en pie de igualdad. Para ello definimos secuencias condensadas de *labels* agrupando los *labels* en bloques consecutivos de la misma clase y considerando cada uno de estos bloques como una única observación del *label*. De esta manera resultan secuencias de *labels* sin repeticiones consecutivas de una misma clase, para las que reservaremos la denominación de “secuencias”, para diferenciarlas de las “series temporales” de *labels* de donde provienen.

Con el fin de entender cómo dependen los *labels* en la secuencia con la velocidad v con la que gira el cilindro del *rotarod*, dividimos las velocidades en tres regímenes, cada uno con cantidades similares de observaciones experimentales. Estos son: velocidades bajas ($v < 13$ rpm), velocidades medias ($13 \leq v < 23$ rpm) y velocidades altas ($v \geq 23$ rpm).

Este análisis nos permitió observar que algunos *labels* aumentan su probabilidad de ocurrencia en la secuencia con la velocidad del rotarod, por ejemplo los *labels* 2, 5 y 7. En cambio,

otros *labels* como el 1, 4 y 9 disminuyen su probabilidad (Figura 2.4b). Estos resultados indican que los animales pasan de una estrategia de locomoción parcialmente sincronizada a bajas velocidades, a una locomoción basada en la alternancia de los miembros inferiores a altas velocidades.

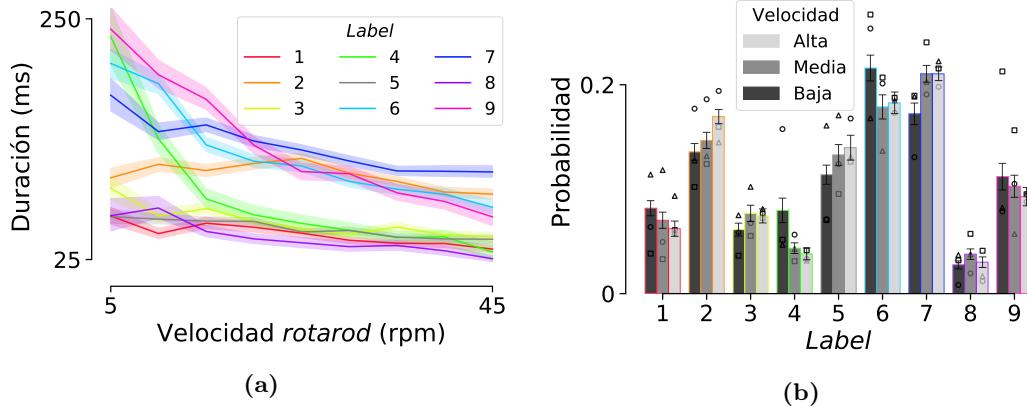


Figura 2.4: Caracterización de la estadística de las secuencias de *labels* y su dependencia con la velocidad del *rotarod*. (a) Duración promedio de cada *label* en la secuencia, con sus intervalos de confianza. (b) Probabilidad de ocurrencia de cada *label* en la secuencia, bajo tres regímenes de velocidad. Las barras de error representan la desviación estándar del promedio de las probabilidades al estudiar los *trials rotarod* por separado. Las probabilidades para cada ratón por separado se indican con marcadores vacíos de color negro (círculos: ratón 1, triángulos: ratón 2 y cuadrados: ratón 3).

Por último nos interesa investigar la dinámica de las secuencias de *labels*, es decir, qué transiciones de poses ocurren con más frecuencia y cómo cambian estas transiciones con la velocidad del *rotarod*. Esperamos dilucidar de esta manera patrones de locomoción de los ratones durante la ejecución de la tarea.

En primera instancia podemos construir un grafo de transiciones a primer orden, esto es, transiciones entre *labels* inmediatamente consecutivos en la secuencia. La Figura 2.5a muestra que las transiciones entre los *labels* $5 \rightleftharpoons 7$, $6 \rightleftharpoons 2$, $5 \rightleftharpoons 2$ y $6 \rightleftharpoons 7$ son las más relevantes en nuestro caso. Casualmente, todas estas son transiciones entre un *label* asociado con dar un paso (2 y 7) y *labels* asociados a poses simétricas (5 y 6). Cabe mencionar que además el *label* 9 (otra pose simétrica) tiene cierta participación, con transiciones destacadas $6 \rightleftharpoons 9$ y $7 \rightleftharpoons 9$.

Cabe destacar que las posiciones en el grafo de los círculos que representan los *labels* se eligieron de manera tal de reflejar las posiciones de los centros de masa de los *clusters* originales del etograma de la Figura 1.5. Esto permite visualizar una cierta jerarquía en los *labels* del mapa t-SNE. Estando los *labels* con transiciones más relevantes (*labels* 2, 5, 6, 7 y 9) ubicados en la parte central del mapa, mientras que los *labels* menos involucrados en las transiciones (*labels* 1, 4, 3 y 8) se encuentran en las periferias del mapa.

Finalmente, la Figura 2.5b ilustra los cambios en las probabilidades de transición al aumentar la velocidad del *rotarod*. La figura representa el cambio en las probabilidades de transición para velocidades altas respecto de las probabilidades de transición a velocidades bajas. Note que la diagonal del *heatmap* es cero, pues no estamos considerando auto-transiciones en la secuencia de *labels* (siempre son transiciones entre *labels* distintos). Se observa un aumento en las ocurrencias de transiciones hacia los *labels* 2 y 7 (esto se aprecia en la figura por las columnas con predominancia de color rojo que aparecen en estos *labels*, indicando aumento de probabilidad a altas velocidades). La figura también indica una disminución en las ocurrencias de transiciones hacia los *labels* 4, 6 y 9 a altas velocidades, como se puede

apreciar debido a las columnas con predominancia de colores azules sobre los mismos. En conjunto, estos resultados indican que a mayor velocidad del *rotarod* el ratón selecciona con mayor frecuencia un tipo de locomoción caracterizada por la alternancia de las patas izquierda y derecha, pasando, en algunos casos, por un estado simétrico intermedio.

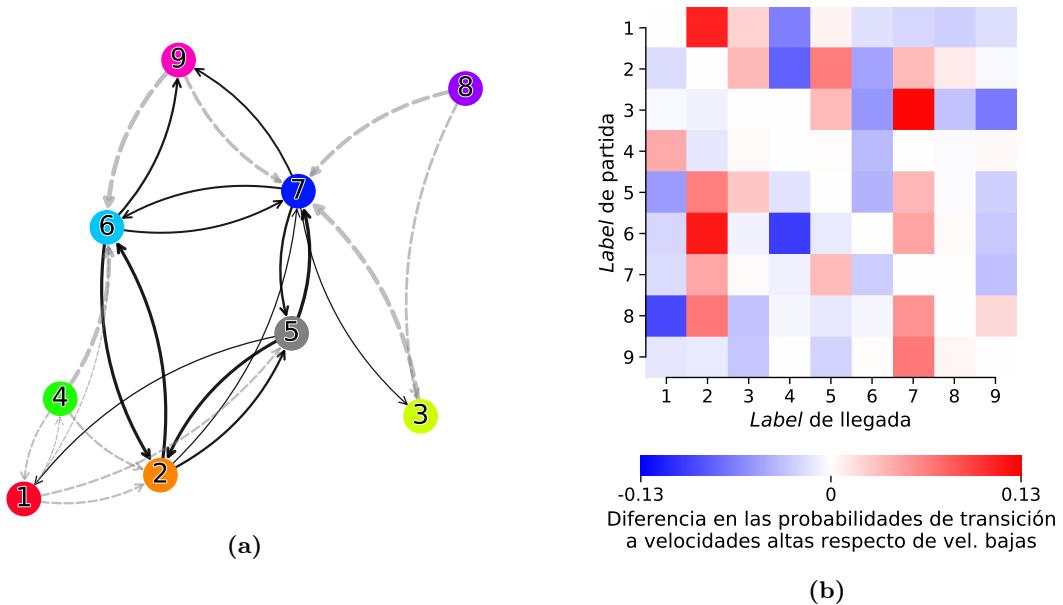


Figura 2.5: Transiciones a primer orden en la secuencia de *labels*. (a) Grafo de transiciones. El grosor de las flechas es proporcional a la probabilidad de transición. Las flechas en línea continua negra representan transiciones que parten desde un *label* con probabilidad marginal mayor a $1/N_{\text{clases}} = 0.11$ en la secuencia (para ver las probabilidades marginales de cada *label* referirse a la Figura 2.4b). Lo opuesto es cierto para las flechas en línea discontinua gris. Solamente se graficaron transiciones que ocurren con probabilidad mayor a $1/(N_{\text{clases}} - 1) = 0.125$. (b) *Heatmap* indicando las variaciones en la probabilidad de transición a altas velocidades respecto de bajas velocidades.

2.4. Dinámica temporal de transiciones entre poses

El análisis anterior solo permite establecer una relación entre *labels* consecutivos. Para lograr analizar la estructura de nuestras secuencias de *labels* a escalas temporales más largas podemos apoyarnos en el uso de logos de secuencias (*sequence logos*). Estos instrumentos de visualización de secuencias fueron desarrollados por Schneider y Stephens en 1990 [34], originalmente utilizados para la visualización de secuencias de ácidos nucleicos (secuenciamiento de ADN y ARN) y de aminoácidos (secuenciamiento de proteínas).

En nuestro caso, podemos utilizar logos de secuencias para visualizar transiciones de orden k entre diferentes *labels*. En particular, vamos a condicionar a estas transiciones alrededor de una determinada clase de *label* $l_0 \in \{1, \dots, N_{\text{clases}}\}$, con $N_{\text{clases}} = 9$. Es decir, nos centramos alrededor de cada uno de los *labels* l que satisfagan la condición $l = l_0$ y calcularemos sus probabilidades de transición de orden k . Más precisamente, la transición de orden $k = 1$ corresponde a calcular las probabilidades de transición desde un determinado *label* $l = l_0$ hacia cualquier otro *label* inmediatamente después, la transición de orden $k = 2$ corresponde a estudiar transiciones 2 pasos en el futuro y así sucesivamente. Por su parte, las transiciones de orden negativo ($k < 0$) corresponden a estudiar transiciones hacia un

determinado *label* $l = l_0$, que provienen desde $|k|$ pasos en el pasado. Identificamos así en un logo de secuencias al orden de transición k con la posición en la secuencia, relativa al lugar donde un *label* adquirió el valor l_0 . Denotaremos además como $l(k)$ al valor del *label* observado en dicha posición relativa k .

Los logos de secuencias son herramientas para graficar el contenido de información de cada posición relativa k de un conjunto de secuencias. En la práctica, para cada clase de *label* $l_0 \in \{1, \dots, N_{\text{clases}}\}$ con $N_{\text{clases}} = 9$, tomaremos todas las secuencias de *labels* observadas centradas en $l(0) = l_0$ y consideraremos transiciones de orden k con $-10 \leq k \leq 10$. Es decir, veremos 10 pasos en el pasado y 10 pasos en el futuro.

Por su parte, el contenido de información es una cantidad proporcional a la entropía de Shannon [24, 34, 35] asociada a la distribución de probabilidad de *labels* $l(k)$ en una determinada posición relativa k de la secuencia, condicionada por haber observado el valor $l(0) = l_0$. Por simplicidad, denotaremos a esta distribución de probabilidad condicional como

$$p_{l',k} = P(l(k) = l' | l(0) = l_0), \quad (2.8)$$

la cual representa la probabilidad de observar el valor de *label* l' en la posición relativa k de la secuencia, dado que observamos el valor de *label* l_0 en $k = 0$.

Estimaremos estas probabilidades a partir de las frecuencias de *labels* observados, es decir

$$p_{l,k} = \frac{N_{l,k}}{\sum_{l'=1}^{N_{\text{clases}}} N_{l',k}}, \quad (2.9)$$

donde $N_{l,k}$ es el total de *labels* con valor l observados en la k -ésima posición relativa del conjunto de secuencias condicionadas por tener el valor de *label* l_0 en $k = 0$.

A partir de estas distribuciones de probabilidad condicionales podemos calcular la entropía de Shannon, en unidades de *bits*, como

$$H(k) = - \sum_{l=1}^{N_{\text{clases}}} p_{l,k} \log_2(p_{l,k}), \quad (2.10)$$

donde $H(k)$ nuevamente es la entropía de Shannon de la posición relativa k condicionada a haber observado el valor de *label* l_0 en $k = 0$.

Finalmente, el contenido de información (eje y) de la k -ésima posición relativa del conjunto de secuencias condicionadas se calcula a partir de $H(k)$ como

$$R(k) = \log_2(N_{\text{clases}}) - (H(k) + e(N_{\text{secuencias}})), \quad (2.11)$$

siendo $e(N_{\text{secuencias}})$ un término de corrección [34] que depende de la cantidad total de secuencias muestreadas $N_{\text{secuencias}}$

$$e(N_{\text{secuencias}}) = \frac{1}{\ln(2)} \frac{N_{\text{clases}} - 1}{2N_{\text{secuencias}}}. \quad (2.12)$$

El valor mínimo que puede adoptar el contenido de información es $R(k) = 0$. Este valor corresponde a tener una distribución de probabilidad uniforme de *labels* en la k -ésima posición, es decir, todos las clases de *labels* tienen la misma probabilidad. El máximo valor que puede adoptar el contenido de información es $R(k) = \log_2(N_{\text{clases}}) = 3.17 \text{ bits}$. Por su parte, este valor corresponde a tener una distribución de probabilidad *labels* completamente determinista, es decir, una única clase de *label* con probabilidad uno y todas las demás con probabilidad cero. Esto ocurre siempre en la posición $k = 0$, pues condicionamos nuestros logos de secuencias a cumplir con $l(0) = l_0$. Por último, el contenido de información satisface que $0 \leq R(k) \leq 3.17 \text{ bits } \forall k$.

Finalmente, en la gráfica de un logo de secuencias, se le asigna a cada clase de *label* l una fracción del contenido de información $R(k)$ en proporción a su probabilidad condicional $p_{l,k}$

$$R_{l,k} = p_{l,k} R(k). \quad (2.13)$$

La Figura 2.6 muestra los logos de secuencias condicionales, centradas alrededor de cada uno de los *labels* obtenidos de la segmentación del mapa de comportamientos. Para cada posición relativa en la secuencia, la clase de *label* más probable se coloca en la base y la menos probable en la cima de la columna. Note que el *label* en la posición relativa central $k = 0$ tiene siempre un contenido de información máximo $R(0) = 3.17$ bits, pues estamos estudiando secuencias condicionadas de esa manera. Note además que el *label* central en $k = 0$ está dentro de un recuadro de línea discontinua y su altura no está a escala. Estos logos de secuencias exponen patrones alternantes recurrentes, que involucran en todos los casos al *label* central, junto con uno o dos *labels* más probables en la alternancia. Puede observarse que los *labels* más involucrados en los patrones alternantes de los demás son los *labels* número 7, 6, 5 y 2.

Además, en la Figura 2.6 pueden observarse oscilaciones en el valor total del contenido de información, especialmente en los logos centrados en los *labels* 6, 7 y 9. Recordemos que el valor de $R(k)$, el contenido de información de la posición relativa k , baja cuando la distribución de probabilidad de los *labels* en esa posición se vuelve más uniforme (probabilidades iguales) y aumenta cuando la distribución se vuelve más concentrada (pocos *labels* mucho más probables que los demás). Por lo tanto, en las oscilaciones del contenido de información, los valles representan posiciones con *labels* indeterminados y los picos están asociados a posiciones donde predominan algunos *labels* sobre los demás.

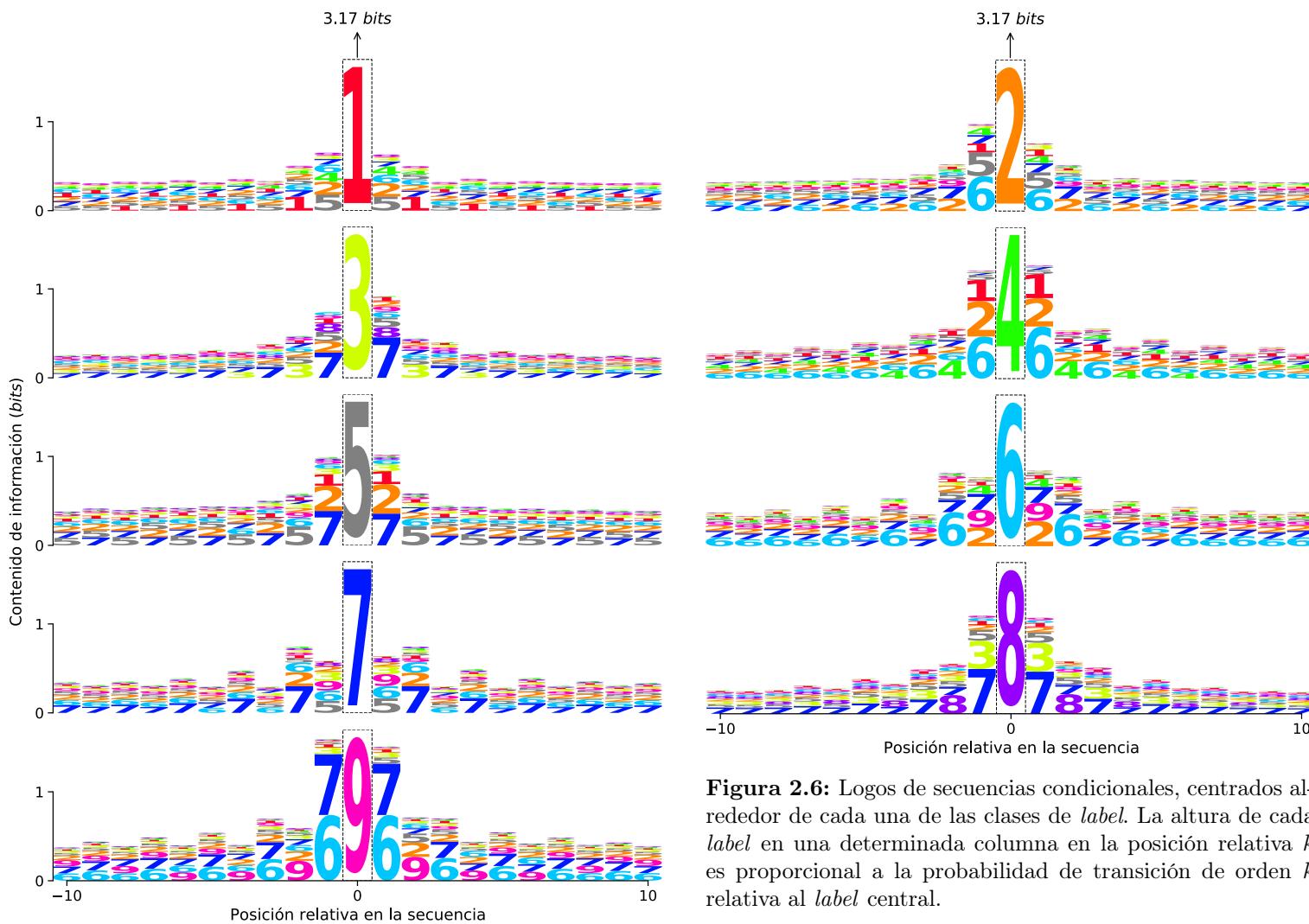


Figura 2.6: Logos de secuencias condicionales, centrados alrededor de cada una de las clases de *label*. La altura de cada *label* en una determinada columna en la posición relativa k es proporcional a la probabilidad de transición de orden k relativa al *label* central.

Capítulo 3

Registros de actividad neuronal

Finalmente, en este capítulo analizaremos los registros de actividad neuronal adquiridos durante la ejecución de la tarea *rotarod* con aceleración. Estos registros electrofisiológicos fueron adquiridos en la región locomotora del mesencéfalo (MLR) de los ratones expertos en la tarea *rotarod*. El MLR es una región del mesencéfalo asociada a los procesos de iniciación y de control de la velocidad en la locomoción [11-13]. Esta región se comunica río abajo con la médula espinal y es necesaria para el proceso de locomoción [36, 37]. Lesiones o patologías en los núcleos de esta región pueden ocasionar ataxia y falta de coordinación en la locomoción [38]. En esta parte del trabajo nos preguntamos si la actividad de algunas de las neuronas presentes en esta región del cerebro estaría modulada por el movimiento del ratón en el transcurso de la tarea *rotarod* [14-16]. Para contestar esta pregunta estudiaremos correlaciones entre la actividad neuronal y la ocurrencia de transiciones de poses.

A continuación, una breve descripción del contenido de este capítulo. Primero, en la Sección 3.1 describiremos el algoritmo de *spike sorting* utilizado para extraer información sobre los tiempos de disparo de las neuronas a partir de las señales registradas por los electrodos. Después, estudiaremos si la actividad de las neuronas registradas está modulada por la ocurrencia de eventos de transiciones de poses. Por un lado, en la Sección 3.2 mostraremos algunos patrones típicos de disparo para neuronas individuales. Por otro lado en la Sección 3.3 mostraremos los patrones de actividad de conjuntos de varias neuronas, ante la ocurrencia de un evento de transición de poses.

3.1. Registro de actividad de neuronas individuales

Se utilizaron implantes crónicos de sondas de silicio (*silicon probes*) conteniendo 16 electrodos para registrar la actividad eléctrica de neuronas del MLR de los ratones durante la ejecución de la tarea *rotarod* (Figura 3.1). La sonda de silicio fue montada sobre una guía que permite cambiar la profundidad de la sonda de manera precisa. Durante la sesión de registro de la actividad neuronal los electrodos fueron conectados a un *headstage* que contenía un conjunto de 16 amplificadores operacionales. A su vez, el *headstage* estaba conectado a un preamplificador de 16 canales [2] que envía la información a la tarjeta de adquisición que sincroniza las señales eléctricas con la información adquirida a través de las cámaras de video.

Se digitalizaron las señales de actividad neuronal a una frecuencia de muestreo de 40 kHz, y se le aplicó un filtro pasa-banda en el rango de 250 Hz a 8 kHz. La señal se pre-procesó utilizando el sistema de adquisición *Neural Recording Data Acquisition System* (Omniplex, Plexon). Finalmente, utilizando el programa *Offline Sorter* (OFS, Omniplex, Plexon) se descompuso la señal electrofisiológica multicanal adquirida en disparos de neuronas individuales.

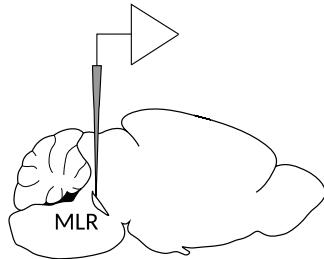


Figura 3.1: Esquema de la ubicación del arreglo de electrodos para el registro electrofisiológico de neuronas en el MLR.

El programa OFS es una implementación de un algoritmo de *spike sorting* supervisado, que identifica *clusters* en las componentes principales de la señal con la actividad de neuronas individuales. Luego, se realizan análisis de correlación cruzada y de auto correlación para corroborar que la identidad de las neuronas identificadas sea única e individual. Es decir, se corrobora que no haya neuronas duplicadas y que no haya neuronas multi-unidad que estén compuestas por las actividades de más de una única neurona [2].

3.2. Actividad de neuronas individuales

Con el fin de comprender si las neuronas del MLR codifican de alguna manera la información de la pose del animal, decidimos realizar un gráfico de raster de la actividad neuronal alineado al inicio de cada pose para cada neurona registrada durante cada ensayo individual. La Figura 3.2 muestra los patrones de disparo de tres neuronas representativas. Habiendo alineado la actividad neuronal a la ocurrencia de un evento determinado, puede estudiarse la distribución de disparos de la neurona en una ventana de tiempo alrededor del evento. Para ello, se utilizó un KDE gaussiano, con un *bandwidth* de 20 ms, para estimar la tasa de disparo (*firing rate*) de las neuronas alrededor de la ocurrencia de un evento. Esto se implementó en Python usando la clase `gaussian_kde` de `scipy.stats` [39]. Esta tasa de disparo se normaliza para que la cantidad total de disparos sea la misma para todas las neuronas. De esta manera, se facilita la comparación de las neuronas de baja actividad con las neuronas de alta actividad. En términos formales, si $f(t)$ denota el *firing rate* de una neurona en torno a un determinado evento, entonces el *firing rate* normalizado $\tilde{f}(t)$ se obtiene de la siguiente manera

$$\tilde{f}(t) = \frac{f(t)}{\int_a^b f(t) dt}, \quad (3.1)$$

donde $[a, b]$ es el intervalo de tiempo estudiado alrededor de la ocurrencia del evento.

En nuestro caso, estudiaremos ventanas de tiempo de $[-500, 500]$ ms alrededor de la ocurrencia de un evento. En particular, consideraremos dos grupos de eventos de transición entre poses: transiciones que salen de un determinado *label l* (evento que denominaremos *offset* del *label l*) y transiciones que llegan a un determinado *label* (*onset* del *label l*).

Además de la tasa de disparo normalizada observada alrededor de la ocurrencia de un evento, la Figura 3.2 muestra también una tasa de disparo de referencia, obtenida luego de intercambiar aleatoriamente los tiempos de disparo de la neurona. Llamaremos a esta tasa de disparo de referencia “*shuffled*”. Dentro de la ventana de tiempo alrededor de la ocurrencia de un evento, la tasa de disparo *shuffled* se calcula permutando los tiempos de disparo de la neurona, bajo la condición de conservar los intervalos de tiempo inter-disparo y el número total de disparos.

Finalmente, en la Figura 3.2 se muestran eventos de transición del tipo *onset* del *label 2*. Recordemos que este *label* está asociado a una pose en la que el ratón da un paso con

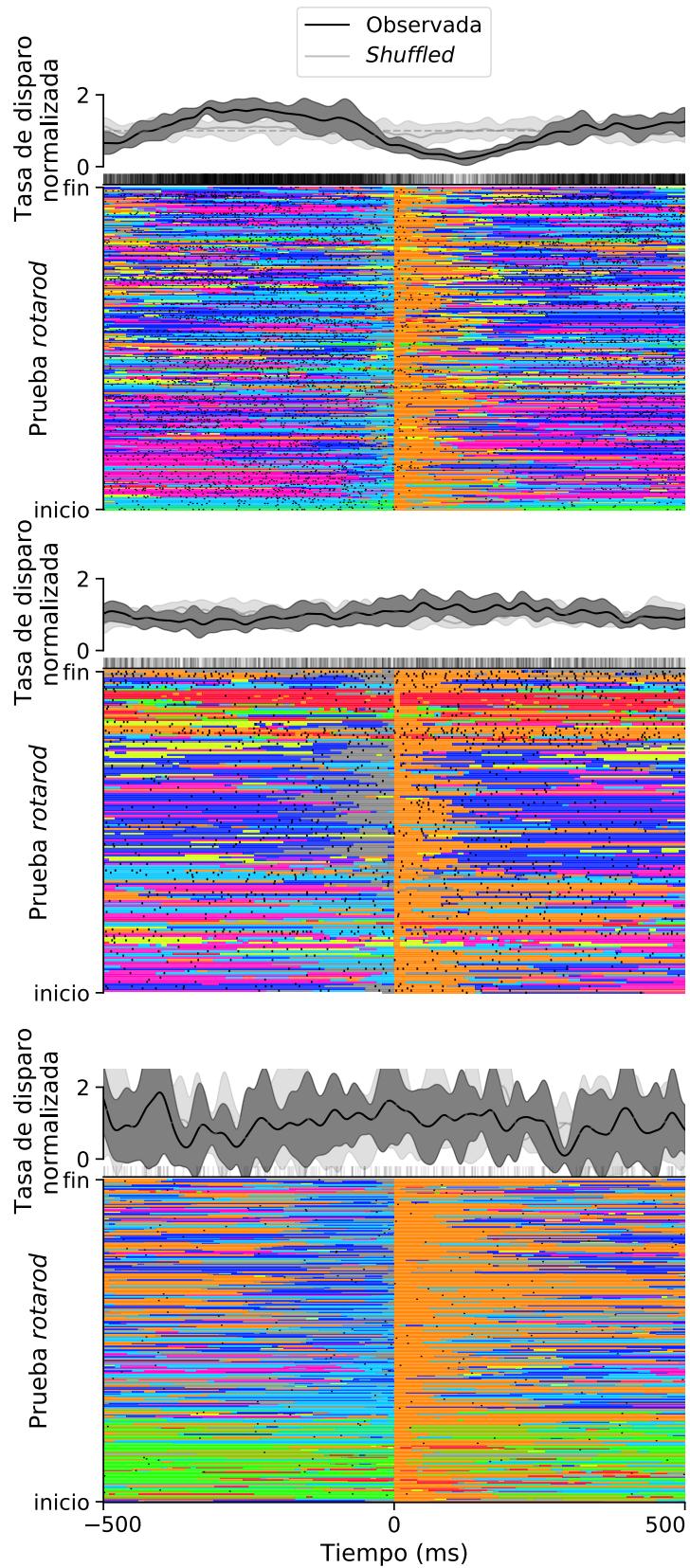


Figura 3.2: Diferentes patrones de actividad neuronal registrados alrededor de un tipo de evento de transición entre poses clasificadas (en este caso, el evento es una transición hacia el *label* 2). Tasas de disparo normalizadas y *raster plots* para tres neuronas diferentes, cada una durante una prueba *rotarod* distinta. Los intervalos de error de las tasas de disparo se obtuvieron mediante *bootstrapping*.

su pata derecha sobre el cilindro *rotarod*. Para cada prueba *rotarod* que se muestra en la Figura 3.2 se utiliza el mismo código de colores del etograma de la Figura 1.5 para indicar la ocurrencia de diferentes *labels*. A su vez, se indican los eventos de disparo de la neurona con marcadores negros (*raster plot*). En la parte superior de cada prueba *rotarod* se muestra el *firing rate* normalizado observado (línea continua negra) y la referencia *shuffled* (línea continua gris). Además se muestra el valor teórico constante de *firing rate* completamente aleatorio y no correlacionado con el evento (línea discontinua gris). Los intervalos de error que se muestran para el *firing rate* se calcularon utilizando un esquema de *bootstrapping* con los tiempos de disparo alrededor de la ocurrencia del evento de transición. Este intervalo de error sirve para comparar la relevancia de la modulación del *firing rate* observado frente al *shuffled*. Por último, puede observarse que algunas neuronas están marcadamente moduladas ante la ocurrencia del evento de transición (Figura 3.2, panel superior), otras están ligeramente moduladas (Figura 3.2, panel medio) y otras neuronas no están moduladas de manera significativa al compararla con la tasa de disparo *shuffled* (Figura 3.2, panel inferior).

3.3. Patrones de actividad de grupos de neuronas

Cabe mencionar que se cuenta con registros de la actividad neuronal a lo largo de diferentes días y cada día consiste en un promedio de 3 a 4 pruebas de *rotarod* por ratón. Con el fin de determinar cuáles neuronas están moduladas por las poses adoptadas por el animal mientras realiza la tarea *rotarod*, tomaremos estos conjuntos de observaciones de una misma neurona en diferentes pruebas de un mismo día y promediaremos sus tasas de disparo en torno a la ocurrencia de un determinado evento de transición. Es importante destacar que consideraremos que los registros de actividad neuronal provienen de diferentes neuronas día tras día, pues la posición del arreglo de electrodos se modificaba cada día alrededor de 50 µm para aumentar el tamaño de la muestra de neuronas registradas.

Para cada una de estas tasas de disparo promedio $\bar{f}(t)$ por neurona, calcularemos el correspondiente *z-score*, definido como:

$$z(t) = \frac{\bar{f}(t) - \mu_{\bar{f}}}{\sigma_{\bar{f}}}, \quad (3.2)$$

donde $\mu_{\bar{f}}$ y $\sigma_{\bar{f}}$ son el valor medio y la desviación estándar de la tasa de disparo promedio $\bar{f}(t)$ para las observaciones de neurona en particular, ante la ocurrencia de un determinado evento de transición entre poses. Por lo tanto el valor del *z-score* $z(t)$ indica cuántos desvíos estándares se aleja la actividad de la neurona en un determinado tiempo t , respecto del valor medio de la actividad de la neurona $\mu_{\bar{f}}$.

Utilizar el *z-score* nos permite comparar la actividad entre diferentes neuronas con facilidad. Por su parte, podemos agrupar neuronas según qué tan moduladas están ante la ocurrencia de un evento. Para determinar cuantitativamente cuándo la actividad de una neurona está modulada o no ante la ocurrencia de un evento de transición se utiliza la prueba de Kolmogorov-Smirnov (*test KS*). Para ello, se utilizó la función `ks_2samp` de `scipy.stats` [39], implementada en Python. A través del *test KS* se comparan las distribuciones del *firing rate* promedio $\bar{f}(t)$ observado con el *shuffled firing rate* promedio de una neurona. Si los valores de ambas tasas de disparo tienen la misma distribución de probabilidad entonces consideraremos que la actividad de la neurona no está modulada por la ocurrencia del evento de transición.

Utilizando el *test KS*, consideraremos que dos muestras de variables aleatorias tienen la misma distribución de probabilidad cuando el valor del estadístico del *test* sea menor a 0.2 y su *p*-valor sea mayor a 0.05. La hipótesis nula del *test KS* es que ambas variables aleatorias provienen de la misma distribución de probabilidad. Además, si solo el valor del estadístico es menor a 0.2 o si solo el *p*-valor es mayor a 0.05 consideraremos que no podemos

rechazar la hipótesis nula. Por lo tanto, ambas variables aleatorias provendrían de la misma distribución de probabilidad. De esta manera, concluiremos que la actividad de la neurona no está modulada por la ocurrencia del evento, dado que los valores que adopta el *firing rate* observado y el *shuffled firing rate* son indistinguibles.

Por el contrario, si se satisface que el estadístico del *test KS* sea mayor a 0.2 y que el p-valor sea menor a 0.05 entonces consideraremos que ambas variables aleatorias tienen distribuciones de probabilidad diferentes. En nuestro caso, esto significa que los valores del *firing rate* observado y del *shuffled firing rate* tienen distribuciones diferentes. Concluyendo, de esta manera, que la actividad de la neurona sí está modulada por la ocurrencia del evento de transición de poses.

Las Figuras 3.3, 3.4, 3.5 y 3.6 muestran los patrones de actividad de grupos de neuronas ante la ocurrencia de eventos de *onset* y *offset* de los *labels* 2, 5, 6 y 7, respectivamente (los patrones de actividad para los *labels* 1, 3, 4, 8 y 9 se muestran en el Apéndice B). Estos patrones de actividad neuronal consisten en *heatmaps* con los valores de *z-score* de cada una de las neuronas observadas. Cada fila en el *heatmap* representa una neurona individual. El tiempo de ocurrencia del evento se indica con una línea discontinua negra vertical. Además, estos *heatmaps* presentan una división entre neuronas no moduladas y moduladas ante la ocurrencia del evento de transición (línea continua negra horizontal).

A su vez, cada uno de estos grupos de neuronas se ordena de tres maneras diferentes para facilitar su visualización: pueden estar ordenados según la posición del máximo o del mínimo de su *z-score* o según un orden por defecto. Al ordenar los *z-scores* de las neuronas no moduladas, según las posiciones de sus máximos o mínimos, esperamos observar un patrón de alineamiento sobre la diagonal del *heatmap* del grupo (esta diagonal se indica con una línea discontinua gris tanto para las neuronas no moduladas como para las neuronas moduladas). Esto se debe a que si las neuronas no están moduladas por la ocurrencia del evento, entonces sus valores de *z-score* serían completamente aleatorios y la posición de su valor máximo o mínimo seguiría una distribución de probabilidad uniforme en la ventana de tiempo analizada. De esta manera, se esperaría que los máximos o mínimos de las neuronas no moduladas estén alineados sobre la diagonal.

Por su parte, se espera que los máximos y mínimos de las neuronas que sí están moduladas presenten desviaciones respecto de la diagonal del *heatmap*, ya que estarían correlacionadas de alguna manera con la ocurrencia del evento de transición. También sería razonable esperar que estas neuronas puedan subdividirse en grupos que responden a la ocurrencia del evento con tiempos característicos similares o con patrones de disparo similares. Cabe mencionar que en nuestro experimento alrededor de un tercio (32 % a 33 %) de las neuronas registradas veían su actividad modulada por transiciones involucrando al *label* 2, mientras que para los *labels* 5, 6 y 7 la fracción de neuronas con actividad modulada fue de un quinto a un cuarto de las neuronas registradas (entre 20 % y 24 %).

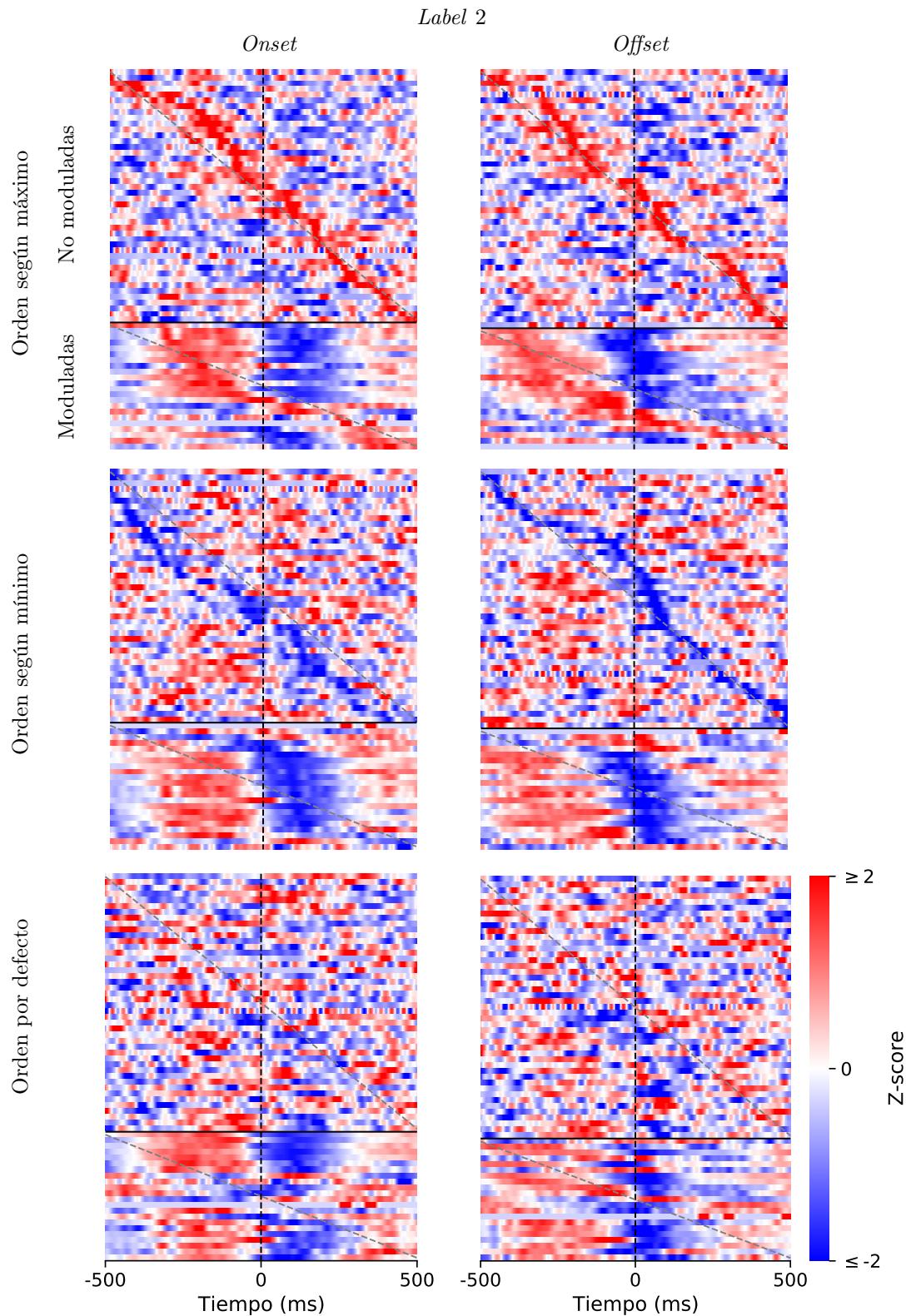


Figura 3.3: Heatmaps con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al *label 2*. Cada fila de un heatmap representa el *z-score* de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.

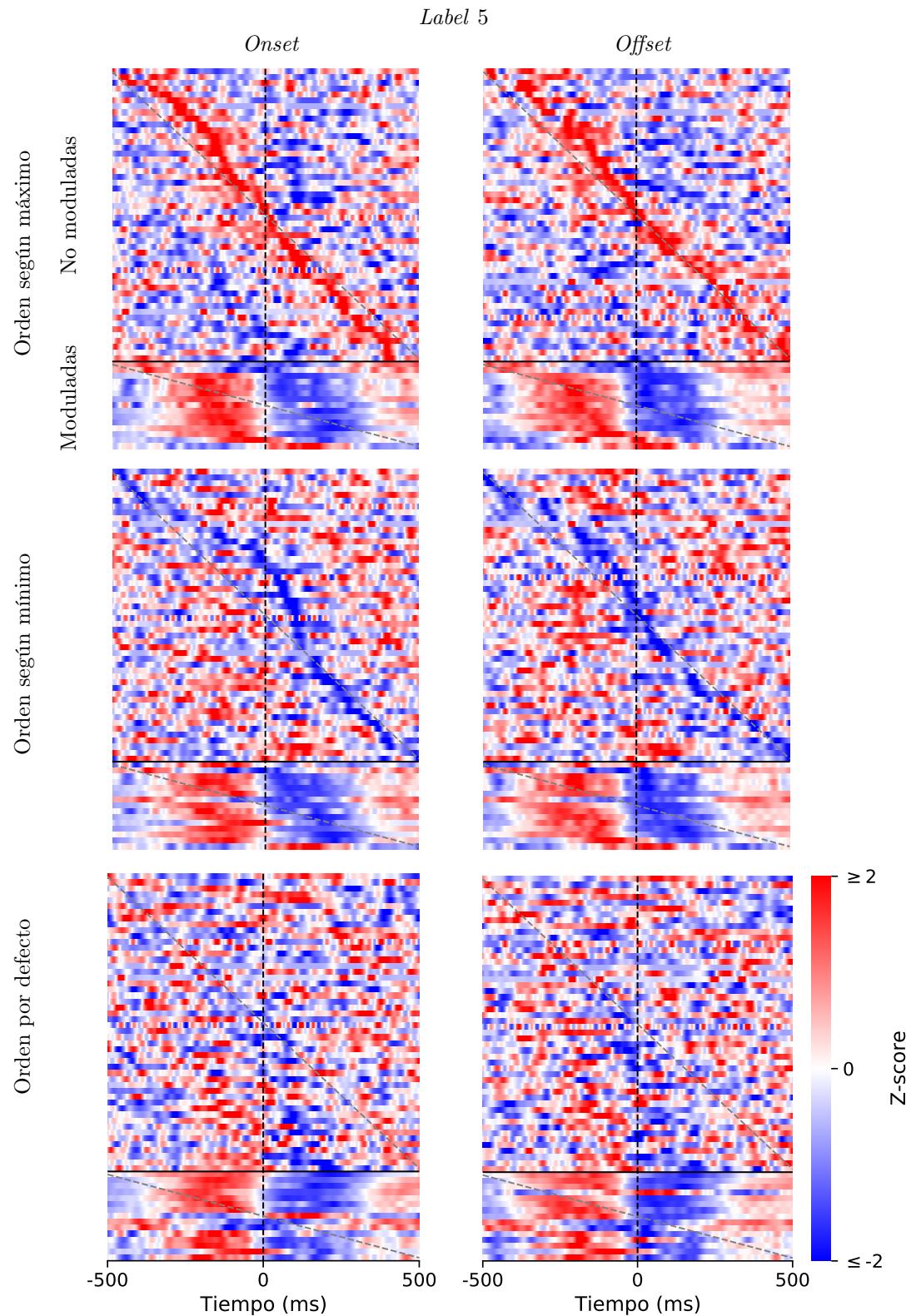


Figura 3.4: Heatmaps con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al *label 5*. Cada fila de un heatmap representa el *z-score* de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.

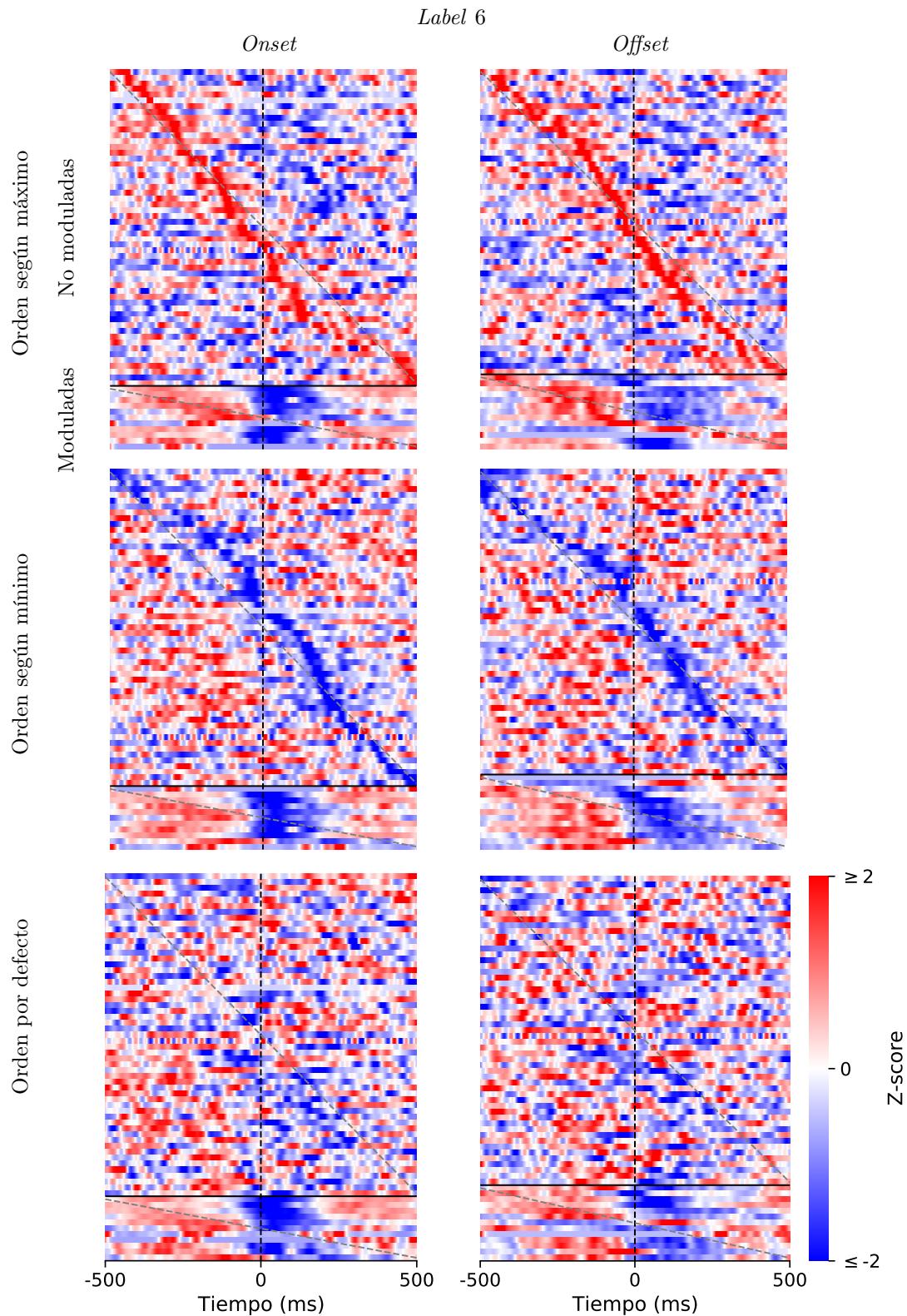


Figura 3.5: Heatmaps con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al *label 6*. Cada fila de un heatmap representa el *z-score* de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.

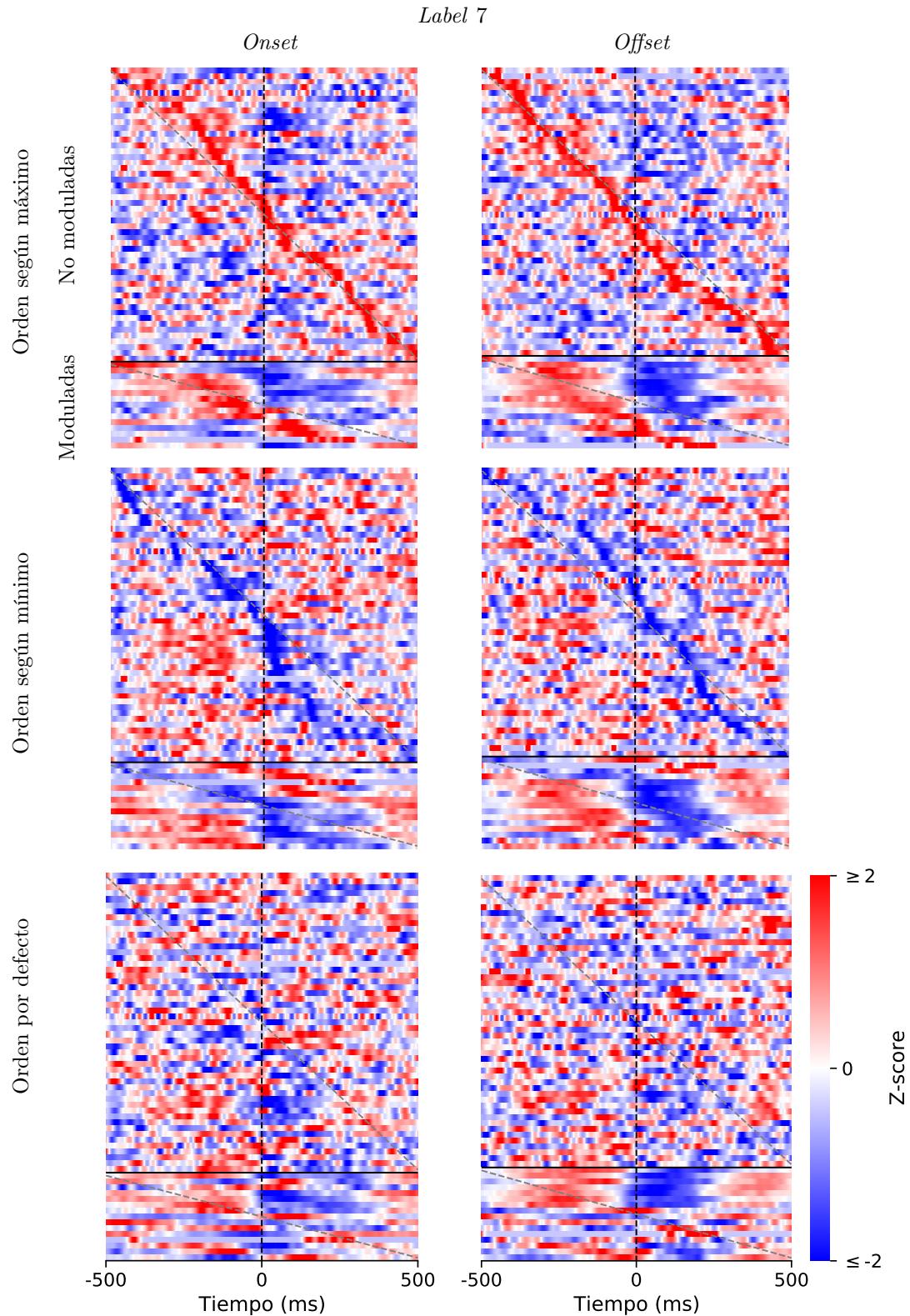


Figura 3.6: Heatmaps con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al *label 7*. Cada fila de un heatmap representa el *z-score* de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.

Conclusiones

En resumen, en este trabajo hemos corroborado que la técnica t-SNE de reducción de la dimensión es apta y ventajosa para la clasificación no supervisada de poses animales durante la ejecución de una tarea motora en el *rotarod* con aceleración. A su vez, vimos que esta clasificación tiene prestaciones que son prometedoras para el estudio del comportamiento animal. En particular, permite automatizar el proceso de clasificación de poses animales, brinda resultados reproducibles y tiene el potencial de cuantificar el comportamiento animal sin la necesidad de que intervenga el sesgo de un experimentador.

Más en detalle, construimos un mapa t-SNE a partir de datos acerca de las posiciones de los ratones y sus partes del cuerpo durante la ejecución de la tarea *rotarod*. Este mapa t-SNE se segmentó utilizando un algoritmo *watershed*, dando origen así a un conjunto finito de clases asociadas a las posiciones de los ratones y sus partes del cuerpo. Se interpretó a este conjunto de $N_{\text{clases}} = 9$ clases como poses específicas (*labels*), que los ratones adoptan durante la ejecución de la tarea motora *rotarod* con aceleración. Más aún, estas poses son representativas a lo largo del conjunto de diferentes ratones con los que se trabajó experimentalmente ($N_{\text{ratones}} = 3$).

La segmentación del mapa t-SNE en regiones discretas fue posible en parte gracias a las restricciones mecánicas que impone la tarea motora estudiada. Esto podría apoyar la hipótesis de emergencia de comportamientos estereotipados producto de restricciones biomecánicas en el movimiento animal.

Además, se corroboró que la duración de las poses clasificadas depende de la velocidad con la que gira el cilindro *rotarod*. Esto es coherente con la intuición de que a mayor velocidad de *rotarod*, mayor es la frecuencia con la que el ratón da pasos sobre el cilindro, por lo que las poses que adopta duran menos tiempo en promedio. Otra característica que se desprende de la clasificación obtenida es que a mayores velocidades de *rotarod* aumenta la probabilidad de transición hacia los *labels* asociados con dar un paso con la patas derecha e izquierda indicando un cambio en el patrón locomotor seleccionado.

Finalmente, observamos cómo son las transiciones a órdenes mayores entre *labels* y estudiamos así la dinámica de transiciones en secuencias de poses. Un resultado interesante es la aparición de patrones alternantes específicos según la clase de pose en la que nos centremos condicionalmente. También se observaron oscilaciones en el contenido de información de las secuencias de poses condicionales. Estas secuencias de poses podrían corresponder a momentos donde el animal se involucra en la ejecución de un patrón de movimiento específico.

Cabe mencionar que en este trabajo capturamos la dinámica del movimiento de los ratones esencialmente en las transiciones entre diferentes poses clasificadas. Otro enfoque posible [1, 40], es la utilización de la transformada de *wavelet Morlet* compleja para obtener, a partir las mismas series temporales de las posiciones relativas de los marcadores de las partes del cuerpo de los ratones ya utilizadas, los espectros de frecuencias que componen estas series temporales. De esta manera esperamos obtener información directa acerca de la dinámica del movimiento de los ratones donde los *labels* que se obtendrían se interpretarían como diferentes estrategias o patrones de movimiento, en lugar de poses específicas [1, 40].

Por último, hemos mostrado el potencial de esta clasificación no supervisada de poses

para estudiar correlaciones con patrones de actividad neuronal. En particular, mostramos la existencia de neuronas (alrededor de un cuarto de las neuronas registradas) en la región locomotora del mesencéfalo cuya actividad se ve modulada por los eventos de transición entre poses. Complementariamente, podría también estudiarse cómo es la actividad de estas neuronas durante el tiempo en el que ocurre un *label* determinado, además de en los momentos en que ocurren transiciones entre *labels* diferentes. La modulación de la actividad neuronal observada sugeriría que el MLR codifica al menos parte de la información involucrada en el movimiento de los miembros inferiores. Sin embargo, la longitud temporal de la modulación observada es mayor a la duración de las poses individuales, indicando que el MLR participaría de la selección del patrón de movimiento del animal más que a la pose específica en sí.

El paso siguiente para seguir avanzando en el proyecto es aplicar el conocimiento adquirido para estudiar cómo cambian las estrategias de movimiento o las transiciones entre poses durante el aprendizaje de la tarea *rotarod*. Esperamos que los comportamientos clasificados gracias a la técnica t-SNE nos permitan cuantificar detalladamente el desempeño durante el aprendizaje de una tarea motora y así evaluar el repertorio de estrategias de movimiento exitosas o ineficientes durante dicho proceso. Por último, esperamos que la técnica t-SNE de clasificación no supervisada del comportamiento no solo sea capaz de distinguir cambios durante el aprendizaje motor sino también cambios a raíz de impedimentos motores como consecuencia de enfermedades neurodegenerativas. Existe evidencia experimental mostrando una reducción en la latencia a caer en modelos de la enfermedad de Parkinson y de esclerosis lateral amiotrófica [7, 41], sin embargo estos trabajos utilizan solo la latencia de caída como medida del desempeño del animal. Nosotros postulamos que el estudio detallado del patrón de movimiento durante la ejecución de la tarea nos permitirá la detección temprana de impedimentos motores y la comparación de diferencias entre modelos experimentales de enfermedades neurodegenerativas que ayude a avanzar en la compresión de los mecanismos circuitales subyacentes [42].

Bibliografía

1. Berman G. *et al.* Mapping the stereotyped behaviour of freely moving fruit flies. *J. R. Soc. Interface* **11** (2014).
2. Tovote P. *et al.* Midbrain circuits for defensive behaviour. *Nature* (2016).
3. Lavzin M., Levy S. *et al.* Cell-type specific outcome representation in primary motor cortex. *Neuron*. <https://www.biorxiv.org/content/early/2020/03/04/2020.03.03.971077> (2020).
4. van der Maaten L., Hinton G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579-2605 (nov. de 2008).
5. Esposito M. *et al.* Brainstem nucleus MdV mediates skilled forelimb motor tasks. *Nature* (2014).
6. Lehner P. *Handbook of ethological methods* 2.^a ed., 24-27. ISBN: 0521637503 (Cambridge University Press, 1998).
7. Costa R. *et al.* Differential Corticostriatal Plasticity during Fast and Slow Motor Skill Learning in Mice. *Curr. Biol.* **14**, 1124-1134 (2004).
8. Mathis A. *et al.* DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nat. Neurosci.* **21**, 1281-1289. ISSN: 1546-1726 (9 2018).
9. Kobak D., Philipp B. The art of using t-SNE for single-cell transcriptomics. *bioRxiv*. <https://www.biorxiv.org/content/early/2018/10/25/453449> (2018).
10. Korsunsky I. *et al.* Fast, sensitive, and accurate integration of single cell data with Harmony. *bioRxiv*. <https://www.biorxiv.org/content/early/2018/11/05/461954> (2018).
11. Caggiano V. *et al.* Midbrain circuits that set locomotor speed and gait selection. *Nature* **553**, 455-460. ISSN: 0028-0836 (7689 2018).
12. Roseberry T. *et al.* Cell-Type-Specific Control of Brainstem Locomotor Circuits by Basal Ganglia. *Cell* **164**, 526-37. ISSN: 0092-8674 (3 2016).
13. Shik M. *et al.* Control of walking and running by means of electric stimulation of the midbrain. *Biofizika* **11**, 659-666. ISSN: 0006-3029. <http://europepmc.org/abstract/MED/6000625> (1966).
14. Skinner R., Garcia-Rill E. The mesencephalic locomotor region (MLR) in the rat. *Brain Research* **323**, 385-389. ISSN: 0006-8993. <https://www.sciencedirect.com/science/article/pii/0006899384903196> (1984).
15. Sherman D. *et al.* Anatomical Location of the Mesencephalic Locomotor Region and Its Possible Role in Locomotion, Posture, Cataplexy, and Parkinsonism. *Frontiers in neurology* **6**, 140. ISSN: 1664-2295 (2015).
16. Le Ray D. *et al.* *Progress in Brain Research* 51-70 (Elsevier, 2011).

17. Luft A., Buitrago M. Stages of Motor Skill Learning. *Mol. Neurobiol.* **32**, 205-216. ISSN: 0893-7648 (3 2005).
18. Cardin J. *et al.* Targeted optogenetic stimulation and recording of neurons in vivo using cell-type-specific expression of Channelrhodopsin-2. *Nat. Protoc.* **5**, 247-254 (2 2010).
19. Savitzky A., Golay M. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry* **36**, 1627-1639. <https://doi.org/10.1021/ac60214a047> (1964).
20. Hinton G., Roweis S. *Stochastic Neighbor Embedding Advances in Neural Information Processing Systems* (eds. Becker S. *et al.*) **15** (MIT Press, 2003), 857-864. <https://proceedings.neurips.cc/paper/2002/file/6150ccc6069bea6b5716254057a194ef-Paper.pdf>.
21. Sidorov G. *Syntactic n-grams in Computational Linguistics* 8-10 (Springer International Publishing, 2019).
22. Yang Z. *et al.* Heavy-Tailed Symmetric Stochastic Neighbor Embedding English. *Advances in Neural Information Processing Systems 22 (NIPS2009)* (2009), 2169-2177.
23. Jost L. Entropy and Diversity. *Oikos* **113**, 363-375. <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.2006.0030-1299.14714.x> (2006).
24. Shannon C. A Mathematical Theory of Communication. *Bell Syst. Tech. J.* **27**, 379-423. <http://dblp.uni-trier.de/db/journals/bstj/bstj27.html#Shannon48> (1948).
25. Cook J. *et al.* Visualizing Similarity Data with a Mixture of Maps *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics* (eds. Meila M., Shen X.) **2** (2007), 67-74. <http://proceedings.mlr.press/v2/cook07a.html>.
26. Kobak D. *et al.* Heavy-tailed kernels reveal a finer cluster structure in t-SNE visualisations. *CoRR* **abs/1902.05804**. <http://arxiv.org/abs/1902.05804> (2019).
27. Linderman G. *et al.* Fast interpolation-based t-SNE for improved visualization of single-cell RNA-seq data. *Nature Methods* **16**, 243-245. ISSN: 1548-7105. <http://dx.DOI.org/10.1038/s41592-018-0308-4> (feb. de 2019).
28. Poličar P. *et al.* openTSNE: a modular Python library for t-SNE dimensionality reduction and embedding. *bioRxiv*. <https://www.biorxiv.org/content/early/2019/08/13/731877> (2019).
29. Pedregosa F. *et al.* Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **12**, 2825-2830 (2011).
30. Meyer F. Topographic distance and watershed lines. *Signal Processing* **38**, 113-125. ISSN: 0165-1684. www.sciencedirect.com/science/article/pii/0165168494900604 (1994).
31. Meyer F., Beucher S. Morphological Segmentation. *Journal of Visual Communication and Image Representation* **1**, 21-46. ISSN: 1047-3203. <http://www.sciencedirect.com/science/article/pii/104732039090014M> (1990).
32. O'Brien T. *et al.* A fast and objective multidimensional kernel density estimation method: fastKDE. *Comput. Stat. Data An.* **101**, 148-160 (2016).
33. Van der Walt S. *et al.* scikit-image: image processing in Python. *PeerJ* **2**, 453. ISSN: 2167-8359. <https://DOI.org/10.7717/peerj.453> (jun. de 2014).
34. Schneider T., Stephens R. Sequence Logos: A New Way to Display Consensus Sequences. *Nucleic Acids Res.* **18**, 6097-6100. <https://alum.mit.edu/www/toms/papers/logopaper/> (1990).
35. Schneider T. *et al.* Information content of binding sites on nucleotide sequences. *Journal of molecular biology* **188**, 415-431 (3 1986).

36. Nielsen J. How we Walk: Central Control of Muscle Activity during Human Walking. *The Neuroscientist* **9**, 195-204. <https://doi.org/10.1177/1073858403009003012> (2003).
37. Capaday C. The special nature of human walking and its neural control. *Trends in Neurosciences* **25**, 370-376. ISSN: 0166-2236. <https://www.sciencedirect.com/science/article/pii/S0166223602021732> (2002).
38. Hathout G., Bhidayasiri R. Midbrain Ataxia: An Introduction to the Mesencephalic Locomotor Region and the Pedunculopontine Nucleus. *AJR. American journal of roentgenology* **184**, 953-6 (abr. de 2005).
39. Jones E. *et al.* SciPy: Open source scientific tools for Python 2001-. <http://www.scipy.org/>.
40. Marshall J. *et al.* Continuous Whole-Body 3D Kinematic Recordings across the Rodent Behavioral Repertoire. *Neuron*. ISSN: 0896-6273. <https://www.sciencedirect.com/science/article/pii/S0896627320308941> (2020).
41. Campos F. *et al.* Rodent models of Parkinson's disease: beyond the motor symptomatology. *Frontiers in Behavioral Neuroscience* **7**, 175. ISSN: 1662-5153. <https://www.frontiersin.org/article/10.3389/fnbeh.2013.00175> (2013).
42. Alfieri J. *et al.* Early Cognitive/Social Deficits and Late Motor Phenotype in Conditional Wild-Type TDP-43 Transgenic Mice. *Frontiers in Aging Neuroscience* **8** (dic. de 2016).
43. van der Maaten L. Accelerating t-SNE using tree-based algorithms. *J. Mach. Learn. Res.* **15**, 3221-3245 (1 2014).
44. Barnes J., Hut P. A hierarchical O(N log N) force-calculation algorithm. *Nature* **324**, 446-449 (dic. de 1986).

Apéndice A

Implementación y detalles técnicos del algoritmo t-SNE

A.1. Métodos aproximados

En principio, el cálculo exacto del gradiente en cada iteración significa la simulación de un problema de N cuerpos que interactúan entre sí (por qué no, mediante resortes). Por lo tanto, la complejidad computacional del algoritmo crece como $O(N^2)$, lo cual limitaría fuertemente la cantidad de puntos que puedan procesarse en la práctica a $N \lesssim 10^4$ [27]. Por este motivo es que se investigan diversas maneras de implementar t-SNE utilizando métodos aproximados que lo hagan más eficiente.

Una de estas aproximaciones concierne al término atractivo del gradiente de la expresión (1.18). Esta aproximación consiste en despreciar las similaridades $p_{j|i}$ que provengan de pares de puntos separados una distancia mayor a $3\sigma_i$. Estas $p_{j|i}$ representan una contribución prácticamente nula a la KLD y pueden ser ignoradas. Además, debido a la manera en que se calcula cada σ_i de manera tal que la perplexidad \mathcal{P} se mantenga constante, es razonable computar solamente los $\lfloor 3\mathcal{P} \rfloor$ vecinos más cercanos de cada punto del conjunto de datos. De esta manera, al comienzo del algoritmo t-SNE se calcula por única vez una matriz de afinidad o similaridad \mathcal{M} (*affinity o similarity matrix*), de tamaño $N \times N$, dispersa, que consiste únicamente de aquellos valores $p_{j|i}$ no nulos [43].

Otra aproximación consiste en realizar un cálculo aproximado de vecinos cercanos con un rendimiento similar al cálculo exacto [27]. Anteriormente, los vecinos más cercanos se calculaban utilizando estructuras de datos llamadas árboles *vantage-point* (*VP trees*), con complejidad computacional $O(N \log N)$ [43]. Reemplazar este método de *VP trees* por uno aproximado reduce la complejidad computacional.

Por otro lado, existen aproximaciones que conciernen al cálculo del término repulsivo de la expresión (1.18), que por el momento continúa siendo equivalente al cálculo del problema de N cuerpos. Una de ellas es la aproximación desarrollada por Barnes y Hut en 1986, originalmente con aplicaciones en física de partículas y astrofísica [43, 44]. La aproximación de Barnes-Hut consiste en particionar el espacio de salida del mapeo t-SNE en celdas regulares utilizando un árbol cuaternario (*quadtree*), como se ilustra en la Figura A.1a. En cada nodo del *quadtree* el algoritmo decide si el centro de masa de la celda puede utilizarse en reemplazo de todos los puntos individuales contenidos en ella o no. La condición para que una celda pueda aproximar a los puntos contenidos en ella es

$$\frac{\mathbf{r}_{\text{celda}}}{\|\mathbf{y}_i - \mathbf{y}_{\text{celda}}\|^2} < \theta, \quad (\text{A.1})$$

donde $\mathbf{r}_{\text{celda}}$ es la longitud de la diagonal de la celda, $\mathbf{y}_{\text{celda}}$ es la posición del centro de

masa de la celda y el parámetro $0 \leq \theta \leq 1$ permite controlar el grado de aproximación. Note que $\theta = 0$ se corresponde con calcular todas las interacciones de manera exacta (sin aproximación). Usualmente se utiliza un valor $0.2 \leq \theta \leq 0.8$, estando los valores mayores de θ asociados a aproximaciones más inexactas, pero más rápidas. En resumen, la construcción del árbol tiene una complejidad computacional $O(N)$, mientras que el tiempo de búsqueda a lo largo del mismo depende del valor del parámetro θ , pero en promedio es de $O(N \log N)$.

Por último, otra manera de aproximar el cálculo de término repulsivo es generando una grilla de puntos e interpolando a partir de ella, como se muestra en la Figura A.1b. El objetivo de esta aproximación es calcular las fuerzas repulsivas que experimentan los puntos de una grilla sintética y luego interpolar las fuerzas que sufren los puntos $\{y_i\}$ de nuestro mapeo a partir de estos valores. La ventaja de este enfoque es que se ahorran cálculos al evaluar las fuerzas repulsivas solo sobre los puntos de la grilla y no sobre los N puntos del mapeo. De esta manera puede reducirse la complejidad computacional a $O(N)$ [27].

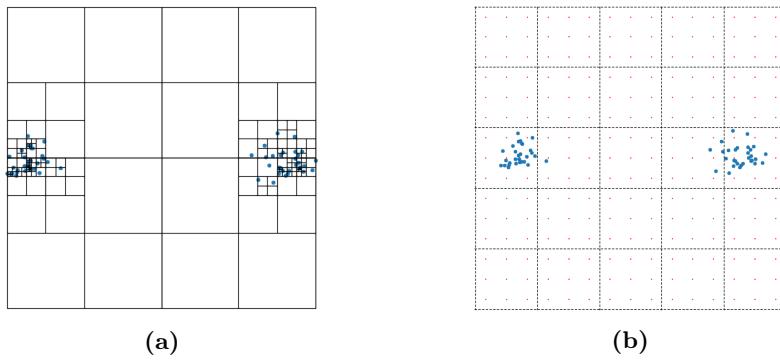


Figura A.1: Métodos aproximados para el cálculo de interacciones entre puntos. (a) Partición obtenida mediante un árbol cuaternario (*quadtree*) (Aproximación Barnes-Hut). (b) Grilla de interpolación (puntos rojos equiespaciados en el plano).

A.2. Técnicas de optimización

Finalmente es importante mencionar que existen diferentes esquemas de optimización de mapeos t-SNE donde el parámetro de exageración $\varepsilon^{(n)}$ varía según la iteración n en la que nos encontremos. En general, las implementaciones de t-SNE utilizan un valor por defecto de $\varepsilon^{(n)} = 12$ en las primeras 250 iteraciones, para luego fijar $\varepsilon^{(n)} = 1$ hasta el final de la optimización [9]. Este procedimiento fue descripto por van der Maaten y Hinton junto con su introducción del algoritmo t-SNE en 2008 [4], y se denomina exageración temprana (*early exaggeration*). Al aumentar drásticamente el valor del factor de exageración, durante las primeras iteraciones del proceso de optimización, se le da un peso importante al término atractivo del gradiente de la función de costo. Para contrarrestar este desequilibrio del término de los p_{ij} (el cual es el término atractivo del gradiente), el término repulsivo responde intentando aumentar los valores de los q_{ij} . Como resultado se fomenta la formación de *clusters* muy densos y separados entre sí en el espacio de salida. En consecuencia, esto crea relativamente mucho espacio vacío en el espacio de salida, lo cual facilita el reacomodamiento de *clusters* durante el resto del proceso de optimización [4].

Dentro del esquema de exageración temprana, suele utilizarse además un esquema de exageración tardía (*late exaggeration*) [9, 27]. Esta vez, se aumenta el valor del factor de exageración durante las últimas iteraciones del proceso de optimización. Esto produce un mapeo t-SNE final con *clusters* más densos y con mayor separación entre sí, facilitando en última instancia la segmentación de los mismos.

La Figura A.2 muestra diferentes mapas t-SNE obtenidos a partir de un conjunto de datos sintéticos, para diferentes valores de parámetro α (diferentes grados de libertad del *kernel* t de Student), según se utilice o no exageración tardía. Todos estos mapas fueron obtenidos utilizando exageración temprana con $\varepsilon = 12$ durante las primeras 250 iteraciones, y el proceso de optimización comprendió un total de 1000 iteraciones. En caso de utilizar además exageración tardía, esta consistió en las últimas 50 iteraciones con $\varepsilon = 2$. Los datos sintéticos consistieron en 10 nubes gaussianas, de 100 puntos cada una, en un espacio de 20 dimensiones, con desviación estándar $\sigma = 1$ y medias aleatorias. Para cada una de estas nubes, 50 puntos se desplazaron en $2\mathbf{e}_{10+l}$ y los otros 50 puntos restantes en $-2\mathbf{e}_{10+l}$, donde l es el número de clase de la nube y \mathbf{e}_j es el j -ésimo vector unitario del espacio de 20 dimensiones. De esta manera se deforman las nubes gaussianas y se les da forma de mancuerna. Note que utilizar un parámetro de grados de libertad $\alpha = 100$ equivale aproximadamente a utilizar un *kernel* gaussiano en el espacio de salida, como se hacía originalmente en el algoritmo SNE. Por otro lado $\alpha = 1$ corresponde implementación usual de t-SNE y $\alpha < 1$ corresponde a distribuciones con colas más pesadas aún. En consecuencia el parámetro α , en conjunto con el esquema de exageración tardía, pueden utilizarse para obtener diversos mapeos t-SNE de un mismo conjunto de datos, según nuestra intención sea la visualización de estos datos o la segmentación de los datos en *clusters* bien definidos.

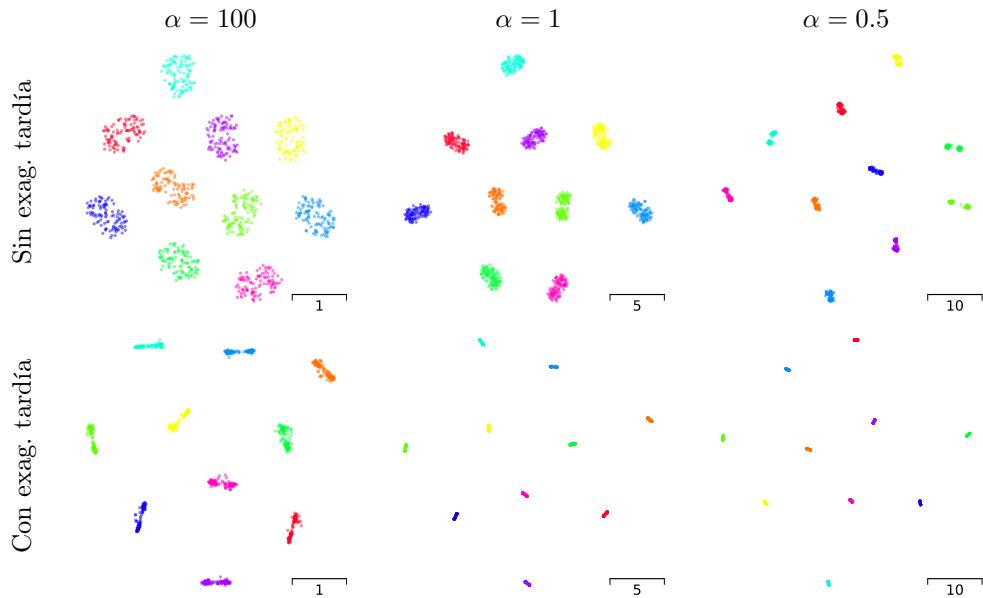


Figura A.2: Comparación de los efectos de utilizar diferentes parámetros o esquemas de optimización del algoritmo t-SNE sobre un conjunto de datos sintéticos: diferentes grados de libertad α en el *kernel* t de Student del espacio de salida del mapeo t-SNE y utilización o no de esquema de exageración tardía durante el proceso de optimización.

A.3. Parámetros utilizados en el mapa de comportamientos

La Tabla A.1 muestra los valores de los parámetros utilizados en el algoritmo t-SNE para la obtención del mapa de comportamientos de la Figura 1.3. Note que el proceso de optimización de un mapa t-SNE puede dividirse en 3 etapas: temprana, regular y tardía, que pueden comprender diferentes cantidades de iteraciones (n_{iter}) y durante las cuales por

lo general varía el valor del factor de exageración (ε). Los demás parámetros del algoritmo t-SNE que no se mencionan en la Tabla A.1 se fijaron en sus valores por defecto en la implementación de openTSNE [28].

Etapa	Exag. (ε)	Iteraciones (n_{iter})	Perp. (\mathcal{P})	Grados de libertad (α)
Temprana	12	250	30	0.4
Regular	5	750	30	0.4
Tardía	10	750	30	0.4

Tabla A.1: Parámetros utilizados durante la ejecución del algoritmo t-SNE mediante el cual se obtuvo el etograma de la Figura 1.3.

Apéndice B

Patrones de actividad de grupos de neuronas, continuación

Las Figuras B.1, B.2, B.3, B.4 y B.5 muestran los patrones de actividad de grupos de neuronas ante la ocurrencia de eventos de *onset* y *offset* de los *labels* 1, 3, 4, 8 y 9, respectivamente. Cada fila en los *heatmaps* representa una neurona individual. El tiempo de ocurrencia del evento se indica con una línea discontinua negra. Además, estos *heatmaps* presentan una división entre neuronas no moduladas y moduladas ante la ocurrencia del evento de transición (línea continua negra). A su vez, cada uno de estos grupos de neuronas se ordena de tres maneras diferentes para facilitar su visualización: pueden estar ordenados según la posición del máximo o del mínimo de su *z-score* o según un orden por defecto.

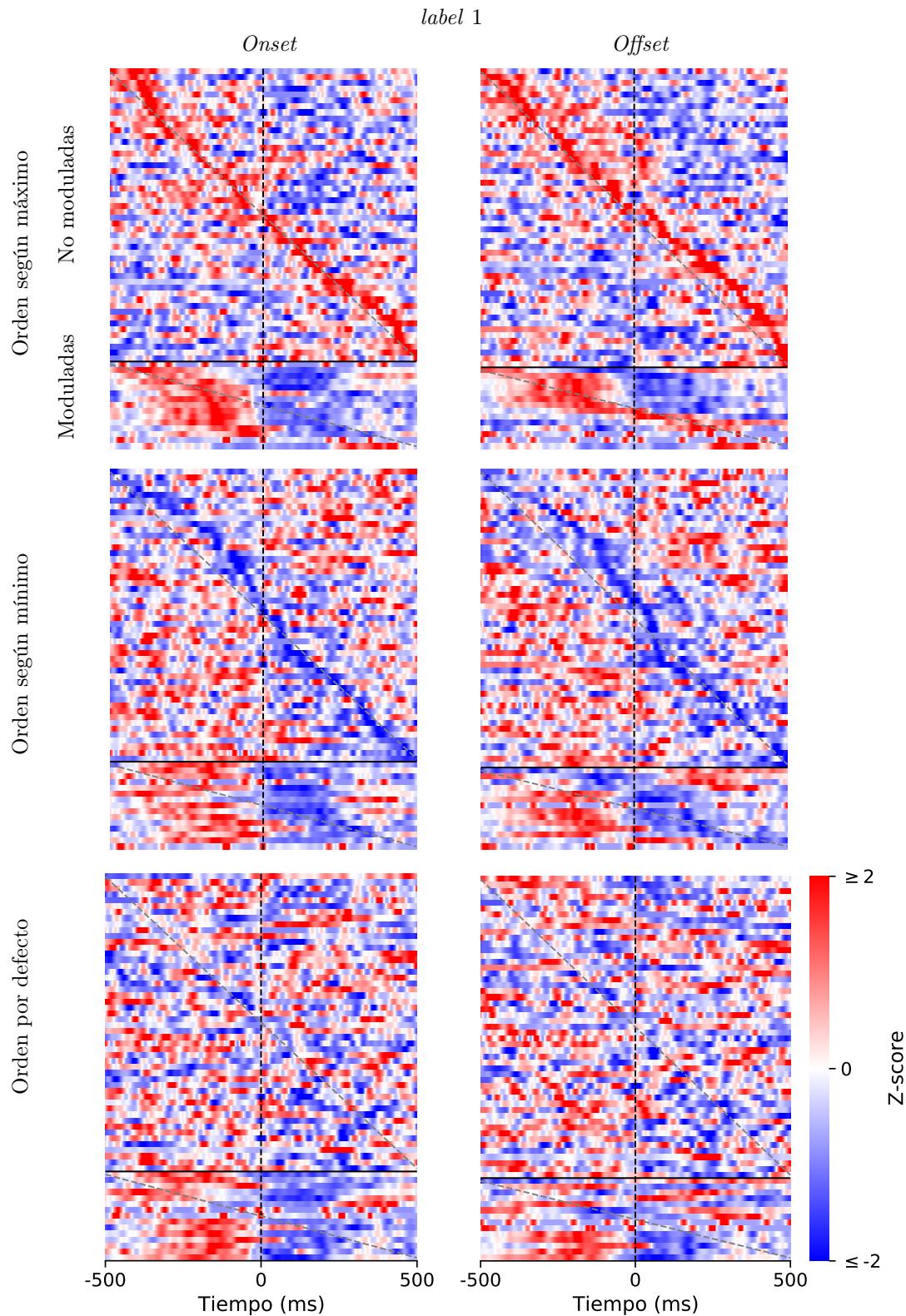


Figura B.1: Heatmaps con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al *label 1*. Cada fila de un heatmap representa el *z-score* de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.

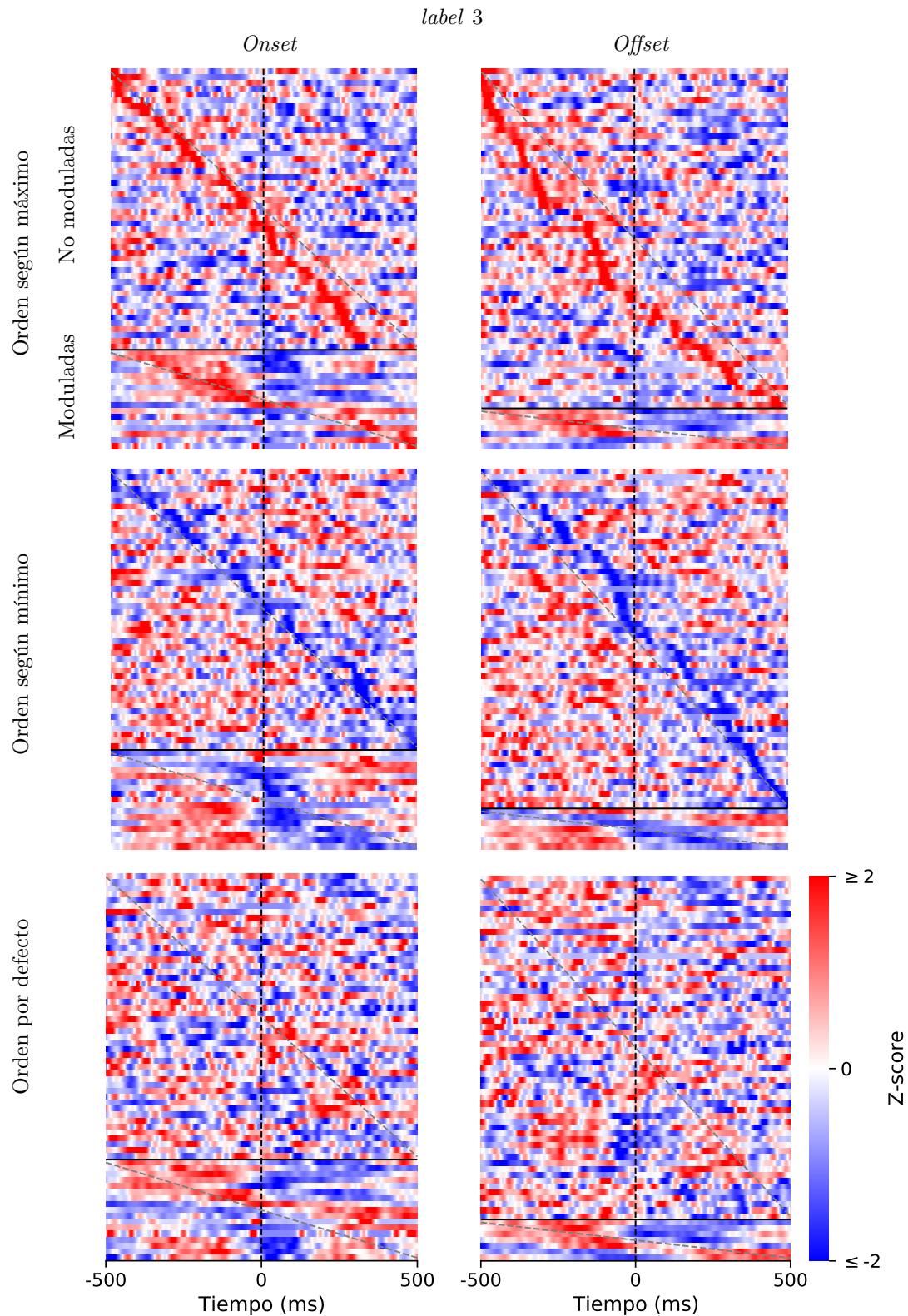


Figura B.2: Heatmaps con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al *label 3*. Cada fila de un heatmap representa el *z-score* de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.

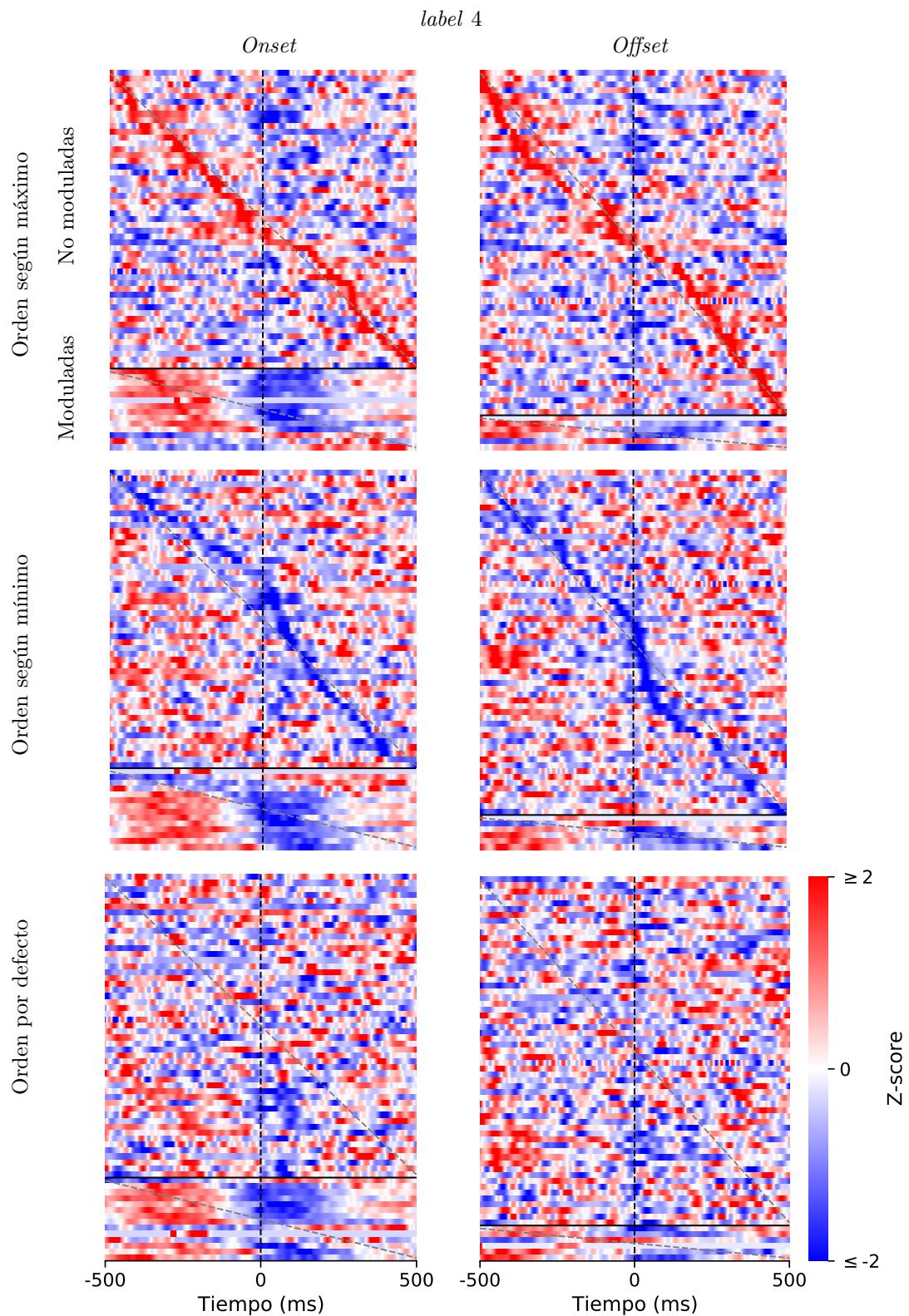


Figura B.3: Heatmaps con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al *label 4*. Cada fila de un heatmap representa el *z-score* de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.

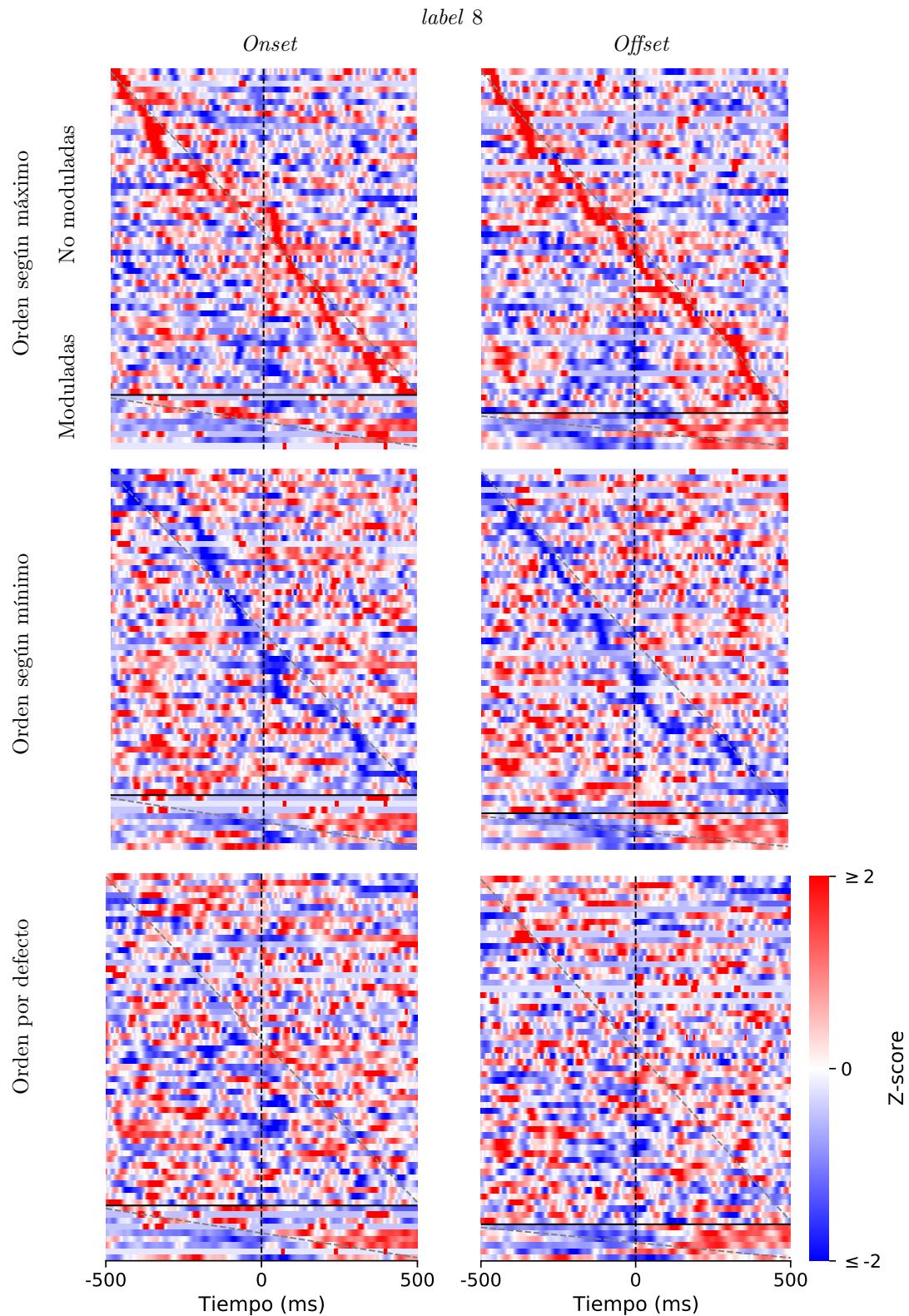


Figura B.4: Heatmaps con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al *label 8*. Cada fila de un heatmap representa el *z-score* de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.

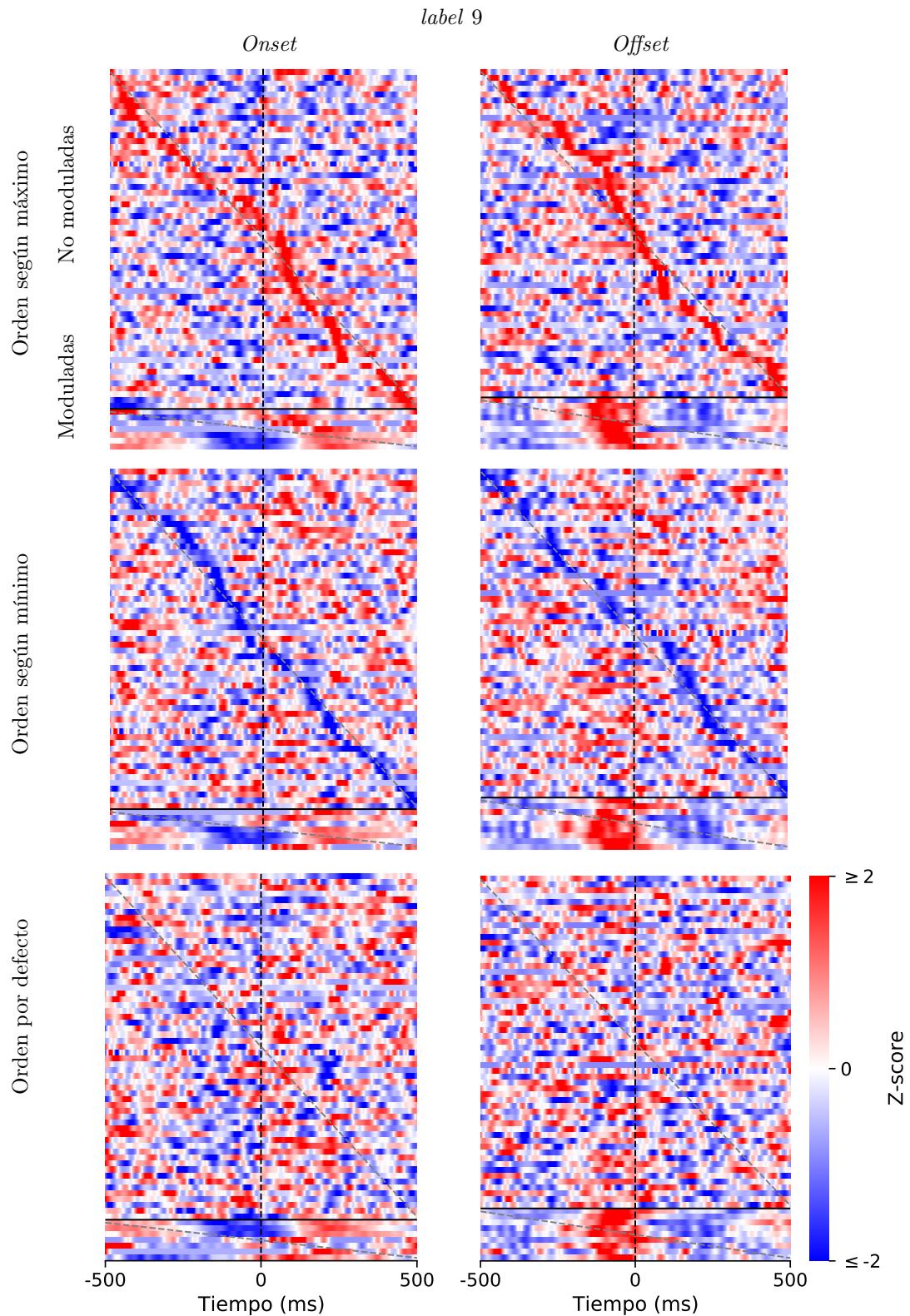


Figura B.5: Heatmaps con las respuestas de grupos de neuronas ante la ocurrencia de eventos de transición que involucran al *label 9*. Cada fila de un heatmap representa el *z-score* de una única neurona. La actividad de las neuronas están divididas en no moduladas y moduladas. Las actividades de las neuronas se ordenan según la posición de su máximo, mínimo y según un orden por defecto.