

Herramientas computacionales en Neurociencias

Departamento de Computación
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Trabajo Práctico Práctica 6 - Spike Sorting

Integrante	LU	Correo electrónico
Daniel J. Foguelman	667/06	dj.foguleman@gmail.com

1. Introducción

1.1. ‘? Por qué Spike-Sorting?

1.2. Temas abarcados en la práctica

1.3. Objetivos

2. Desarrollo

2.1. Datos, análisis

2.2. Filtering

2.3. Feature análisis

2.4. Clustering

3. La Práctica

3.1. Problema 1

En este problema se pide graficar un dataset que tiene la siguiente estructura: `spikes` | `cluster_class` | `inspk` | `par`
Y analizar la cantidad de neuronas que podrían encontrarse en dichos datos.

Comenzamos trabajando sobre la variable `spikes`. Entendemos que estos datos ya se encuentran:

- Filtrados
- Ya se ha hecho un trabajo de Spike detection (con Amplitude Threshold o algún método similar).

El problema de Spike-Sorting, a grandes rasgos, trata de mapear un tren de spikes en múltiples canales de medición con el de una sola neurona. Las dificultades para distinguir cada neurona en particular puede deberse a:

- Distintos canales miden la misma neurona desde distintos lugares y distancias, mostrando distintos valores. (Quin Quiroga cita Gold et al., 2006)
- La sincronización inter-neuronal lleva a la superposición de señales. (Phase Synchronization).
- Mala elección de parámetros en los procesos de Detección de Spikes y de Features.

La pregunta que subyace de este análisis sería entonces ¿cuántas neuronas podemos contar a partir de los datos que tenemos? Sabemos que habrá al menos tantas neuronas como clases de features. ¿Pero cuántas más habrá? ¿Cómo distinguimos neuronas sincronizadas?

En la figura 2 podemos observar cuatro features distintos, ¿cuántos otros features no son analizables a simple vista y precisan de un proceso más sofisticado?

Las preguntas que queremos responder utilizando algún método de Clustering entonces serían:

- Cada Feature corresponde al spike-train de **una** neurona o de varias?
- Hemos distinguido **todos** los features?

- Podemos encontrar señales de otras neuronas en ellos?

Poniendolo en términos de data mining, podemos armar una partición de las N observaciones en K clusters donde cada observación pertenece al cluster con media más cercana. Haciendo un análisis meramente cuantitativo, podemos ver que para $N > K$, por el principio del palomar, dos observaciones pertenecerán al mismo cluster. Esto implica que por cada cluster, estaremos necesariamente encontrando señales de varias neuronas, o bien sincronizadas, o bien que no pudimos separar en la etapa de filtrado o de detección de features.

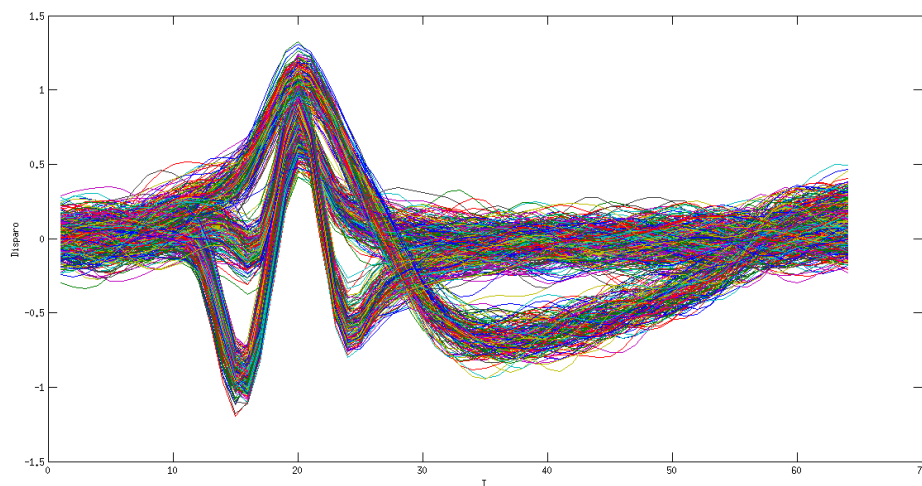


Figura 1: Spikes por tiempo. Datos de data_sim1.

Veamos qué sucede aplicando el método de K-Means a los datos dados:

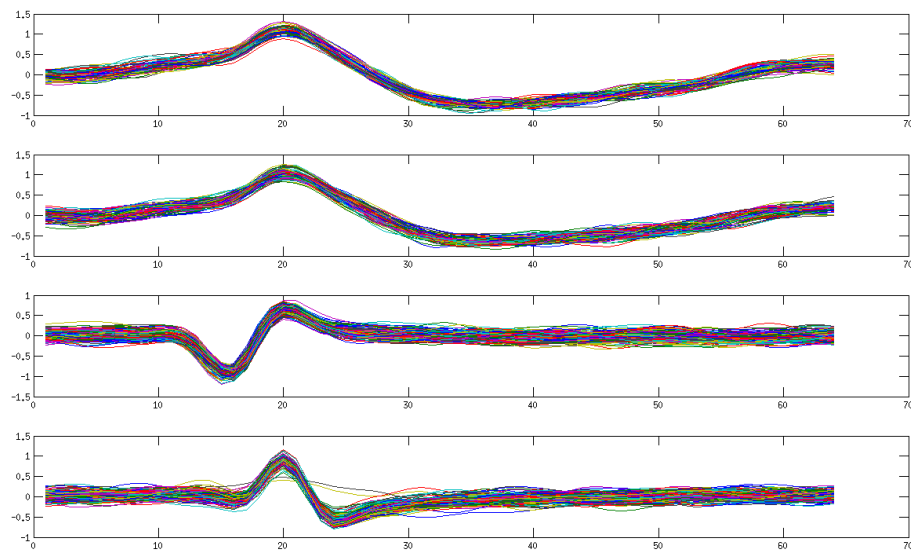


Figura 2: Clustering en cuatro clases los datos en data_sim1.