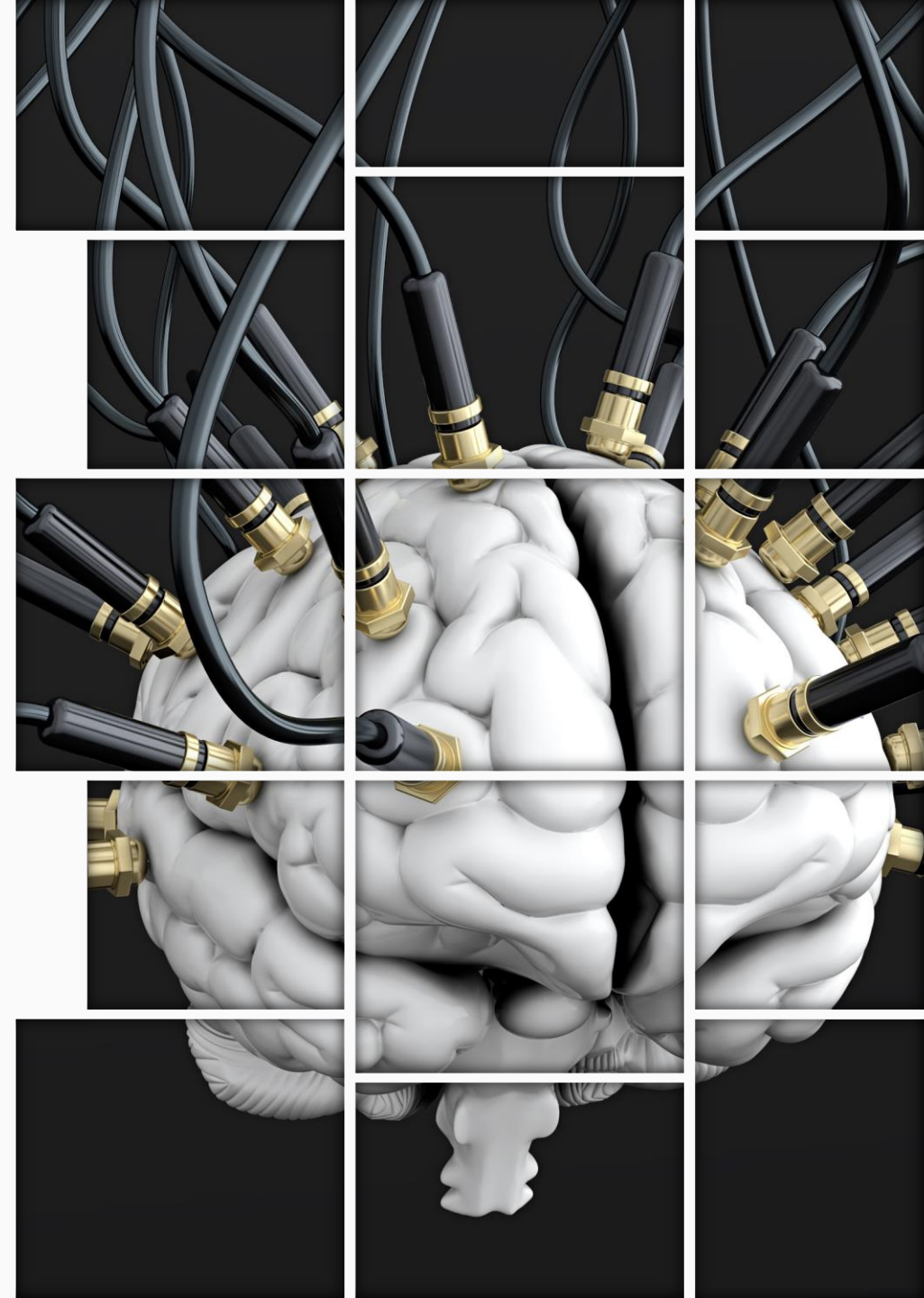
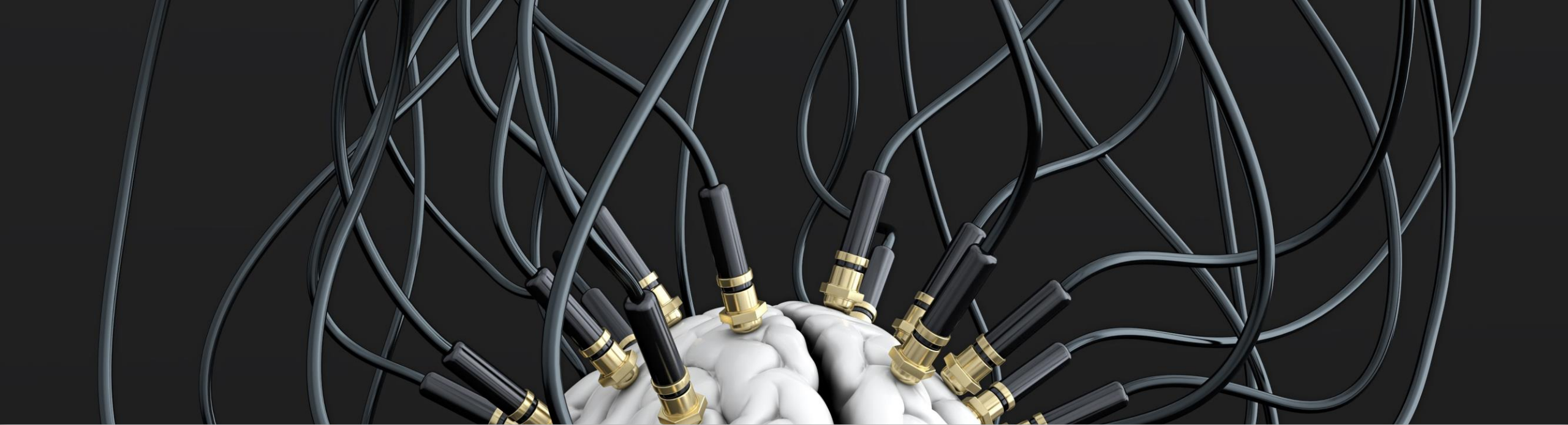


Taller N°4

Introducción a la clasificación de SSVEPs

Mgr. Bioing. Baldezzari Lucas
Profesor Encargado
Ingeniería Biomédica





Objetivos del taller

- Entender por qué son necesarios métodos de extracción de características y clasificación de señales de EEG.
- Introducirnos en algunos métodos de clasificación para Steady State Visual Evoked Potentials (SSVEPs).
- Crear y entrenar un modelo de Convolutional Neural Network (CNN) para clasificar SSVEPs.
- Utilizar la CNN entrenada para clasificar SSVEPs.
- Realizar ejercitación con datos reales.

Extracción de características



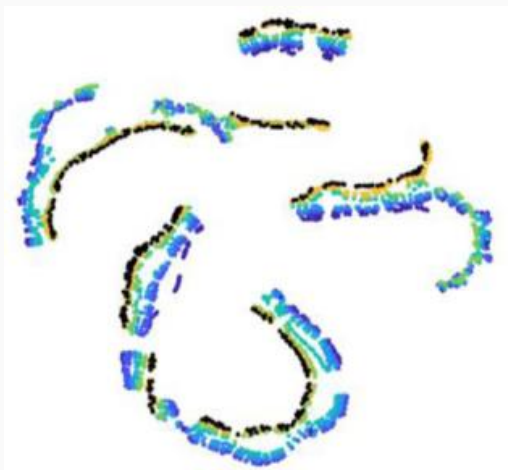
Extracción de características

Obteniendo información de la señal de EEG

El propósito de una ICC es **detectar** y **cuantificar** las **características** inherentes en la señal de EEG, de tal manera de **identificar** cuales son las **intenciones** del del usuario, y así la **traducir** estas intenciones en tiempo real para el **control** o **comando** de un dispositivo y otorgar una retroalimentación adecuada.

Las señales cerebrales para lograr esto se llaman **características** de la señal.

La **extracción de características** es el proceso de distinguir aquellas características relevantes de aquellos contenidos extraños o innecesarios de mi señal de EEG original, para que sean representados de tal manera de que sean interpretados por una persona o una computadora.



Características obtenidas a partir de la magnitud del espectro de Fourier.
Obtenida de [3]



Extracción de características

Obteniendo información de la señal de EEG

Llamamos **características fundamentales** a aquellas que se pueden medir o cuantificar directamente desde la señal. Por si mismas, proveen información limitada.

Las ICC utilizan **características complejas formadas por varias características fundamentales**, las cuales son el resultado de aplicar diferentes técnicas de procesamiento lineales y no lineales, mediciones estadísticas u otros tipos de transformaciones sobre múltiples canales del EEG y/o múltiples puntos temporales y/o espaciales del EEG.

La mayoría de las características utilizadas en aplicaciones de ICC están basadas en análisis espaciales, temporales y/o espectrales de la señal de EEG.

Se extraen características de manera simultanea a partir del EEG, de esta manera forman un set de características que recibe el nombre de **vector de características**.

Estas características, seleccionadas adecuadamente, podrían reflejar las intenciones del usuario de la ICC.



Extracción de características

Obteniendo información de la señal de EEG

Las características extraídas deben cumplir los siguientes puntos,

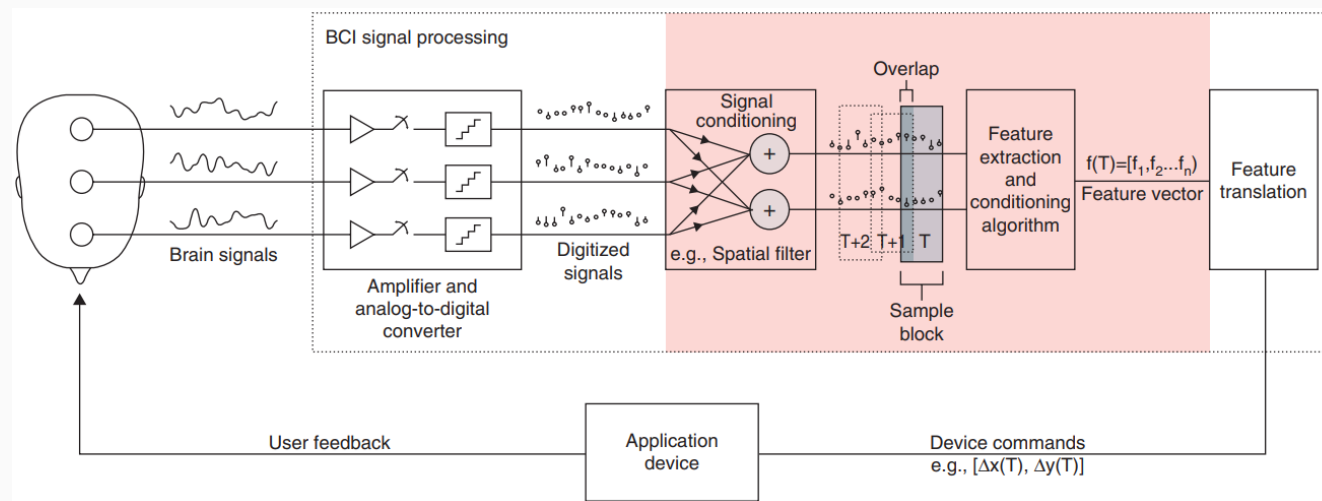
- Las características espaciales, temporales, espectrales y dinámicas deben estar bien caracterizadas para una sola persona o para un grupo de personas.
- Deben poder ser moduladas por el usuario y deben poder ser utilizadas con otras características para lograr transmitir de manera confiable las intenciones del usuario.
- La relación entre la intención del usuario y las características extraídas debe mantenerse estable a lo largo del tiempo y / o se debe poder adaptarse de manera consistente y confiable.

Extracción de características

Obteniendo información de la señal de EEG

Los **tres pasos** para la extracción de características son,

1. Acondicionamiento de la señal para **disminuir el ruido** y **resaltar aspectos relevantes** de la señal de EEG. Importante, recordar que no es lo mismo *ruido* que *artefacto*.
2. Extraer las características a partir de la señal acondicionada.
3. **Acondicionamiento de las características** extraídas para preparar el vector de características antes de su ingreso a al bloque de clasificación y traducción.





Extracción de características

1. Acondicionamiento de la señal

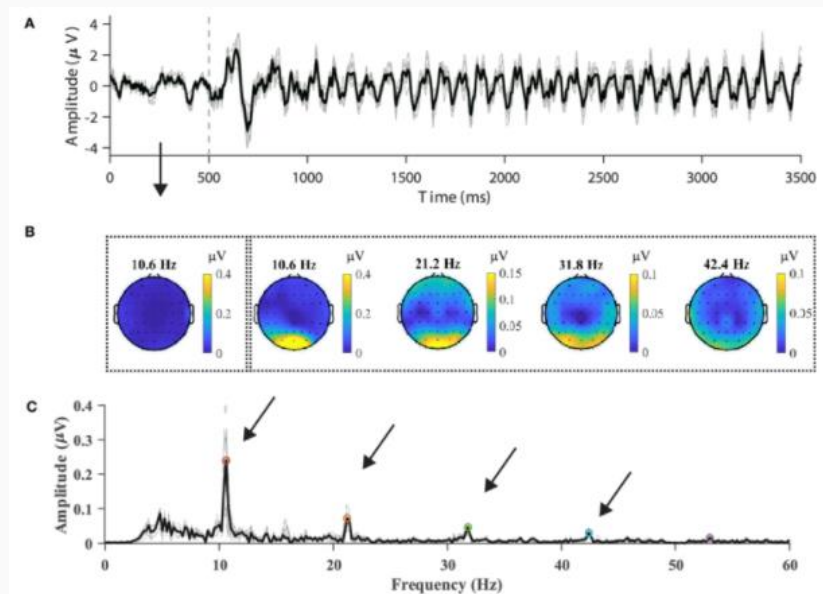
- Filtro pasabanda.
- Filtros espaciales.
 - Filtros independientes de los datos –filtros CAR, Laplacianos-.
 - Filtros dependientes de los datos –filtros PCA, ICA, CSP-.
- Decimación
- Normalización.
- Eliminación de artefactos provenientes de otras fuentes biológicas e interferencias externas.

Extracción de características

2. Extracción de características a partir de la señal acondicionada

En el caso ideal, debemos conocer las características de nuestra señal para seleccionar el mejor **método para extraerlas**.

¿Cuales son las características de los SSVEPs? ¿En qué dominio –temporal o frecuencial- podríamos obtener mejor información?



"BETA: A Large Benchmark Database Toward SSVEP-BCI Application". Yijun Wang. Chinese Academy of Sciences. Junio 2020.



Extracción de características

2. Extracción de características a partir de la señal acondicionada

- Características temporales – detección de picos, integración-.
- Características Frecuenciales (espectro).
 - Band power.
 - FFT.
- Características en tiempo-frecuencia. Se utiliza la transformada Wavelet.
- Características que buscan similitudes espaciales.
 - Coherencia. Un ejemplo en frecuencia es el método Canonical Correlation Analysis.



Extracción de características

3. Acondicionamiento de las características extraídas para preparar el vector de características antes de su ingreso a al bloque de clasificación y traducción.

La distribución y relación entre las características extraídas puede tener un impacto significativo en la performance de nuestro clasificador, para resolver esto, se aplican métodos para normalizar los valores a lo largo de nuestro vector de características.

Comúnmente se logra restando el valor promedio de la señal y escalando la amplitud de la señal para tener varianza unitaria.

Clasificación





Traducción y Clasificación

Introducción

El vector de características nos da información indirecta de las intenciones del usuario.

Debemos usar un algoritmo de **traducción y clasificación**.

Estos algoritmos toman como entrada un vector de características y **generan** una o varias **salidas u outputs** que representan el **set de comandos** que la ICC puede llevar a cabo.

Ejemplo sencillo, modelo lineal,

$$Y = aX + b$$

¿Qué son a y b?

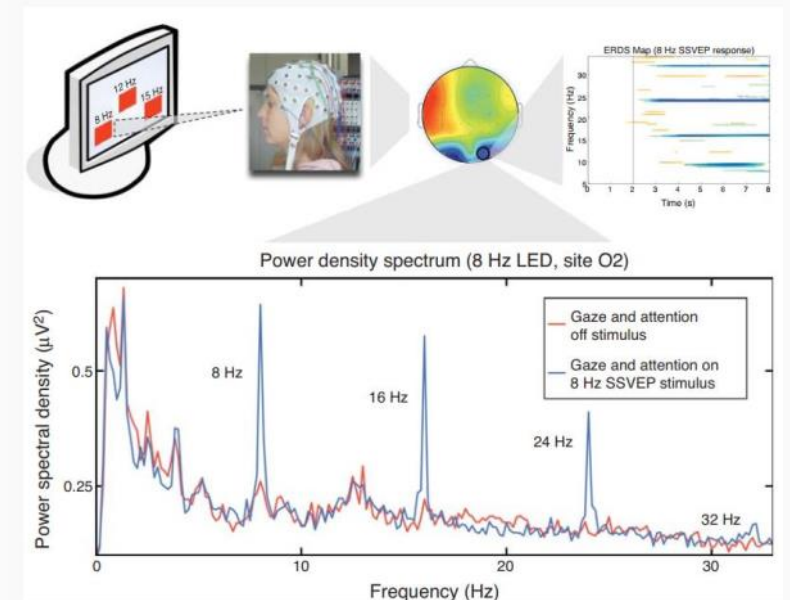
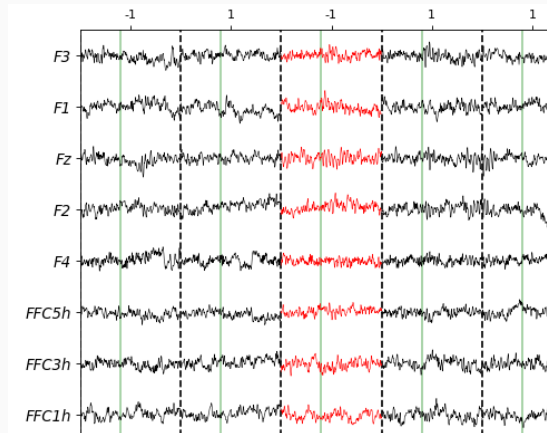
¿Cómo obtenemos los valores asociados a estos parámetros?

Traducción y Clasificación

Introducción

Una vez que seleccionamos el modelo debemos entrenarlo, ¿por qué?, ¿cómo lo hacemos?

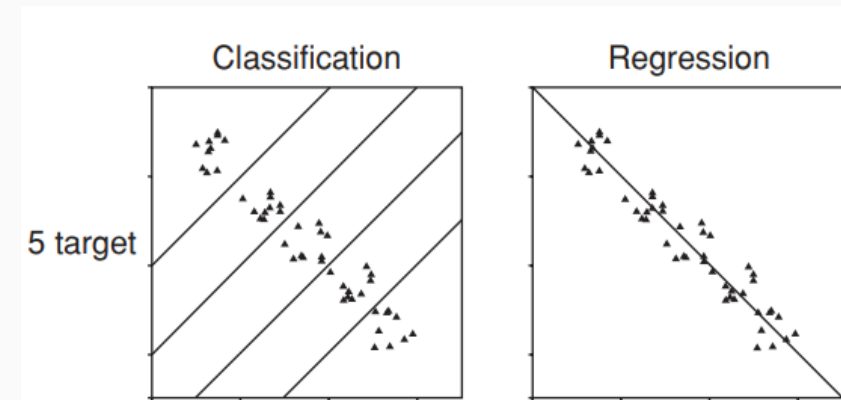
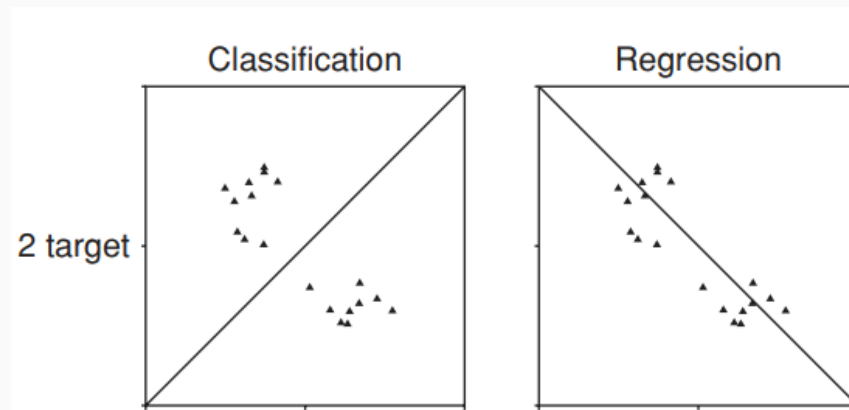
Para entrenar un modelo se utiliza un **set de datos de entrenamiento**, el cual consta de *training samples*, que son nuestros **vectores de características** y de *training labels* –en el caso de aprendizaje supervisado-, que son los **comandos** asociados a esos vectores.



Traducción y Clasificación

Tipos de modelos

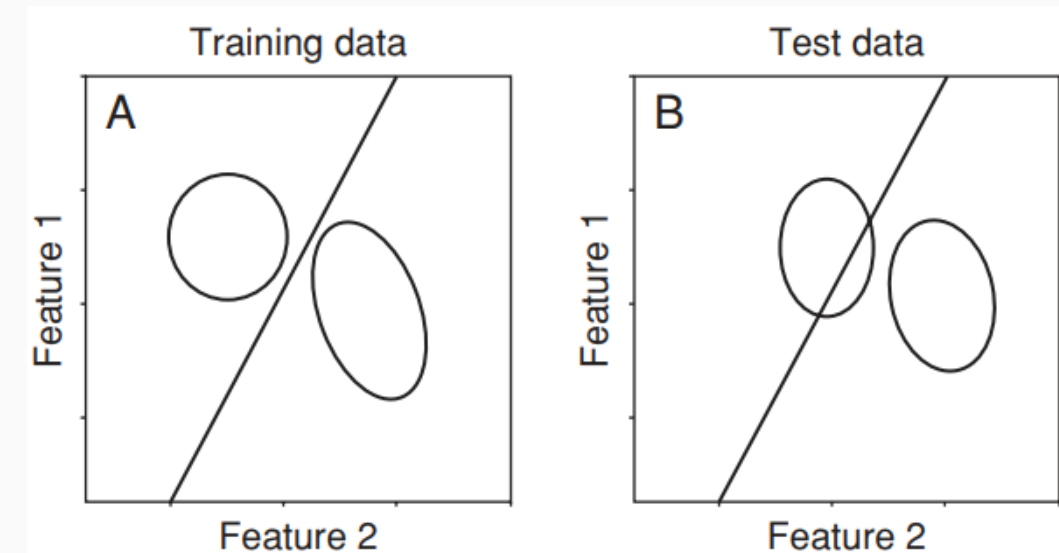
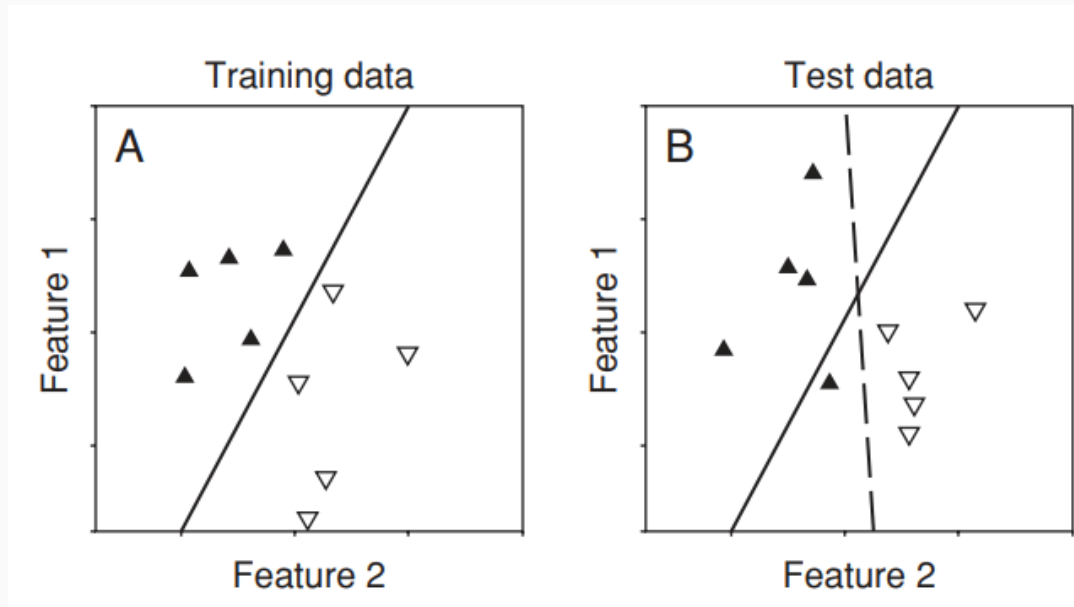
- ✓ Salidas discretas → funciones de discriminación. Este tipo de funciones traducen las observaciones –vectores de características- en salidas discretas, categorías o clases, por ejemplo, diferentes tipos de letras.
- ✓ Salidas continuas → funciones de regresión. Este tipo de funciones traducen las observaciones –vectores de características- en salidas o variables continuas, por ejemplo, la posición del cursor de un mouse sobre una pantalla.



Traducción y Clasificación

Tipos de modelos

Ejemplo: Discriminante lineal o de Fisher.



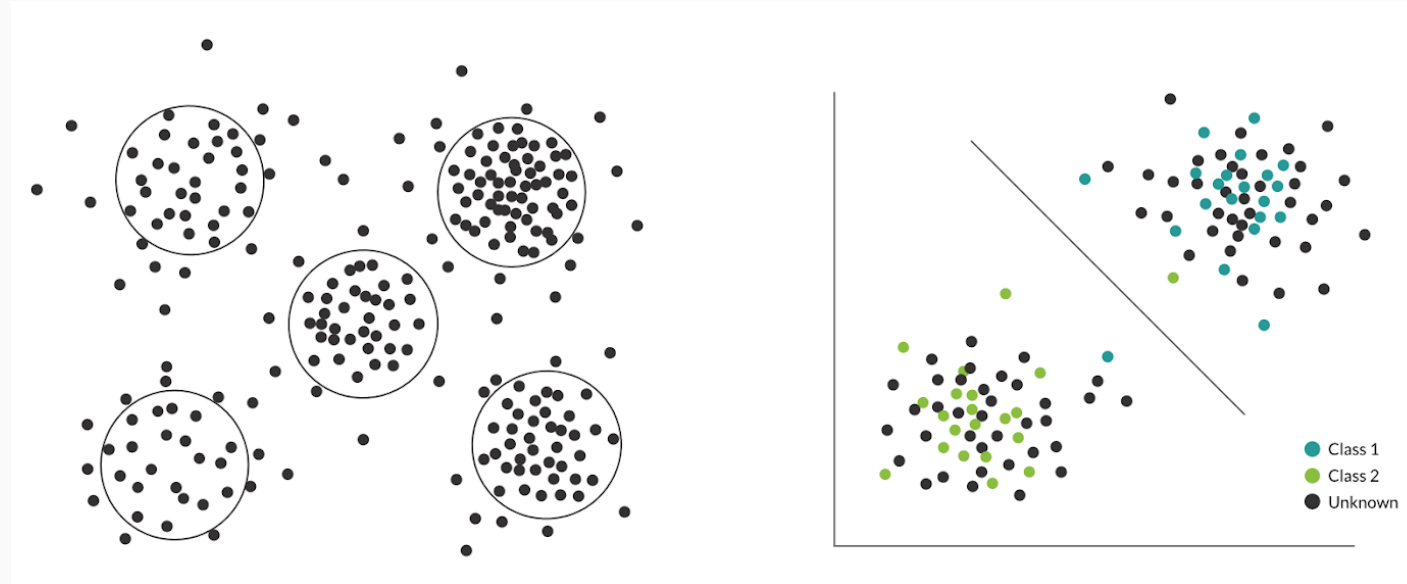
¿Cómo podríamos mejorar la performance de nuestro clasificador para que generalice mejor nuevos datos?

Las características de señal de EEG cambia con el tiempo.

Traducción y Clasificación

Tipo de aprendizaje

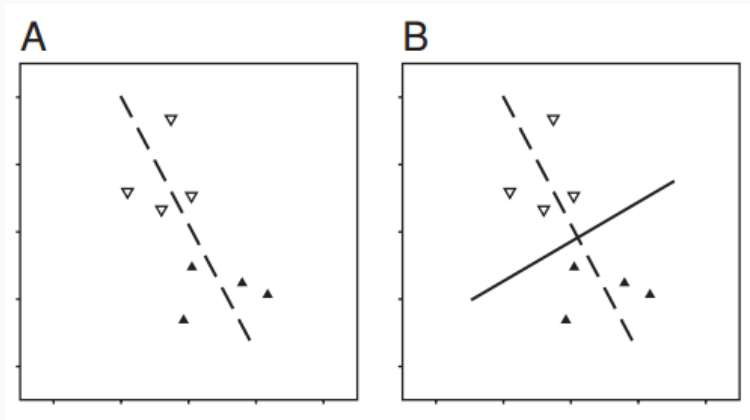
Aprendizaje supervisado vs no supervisado.



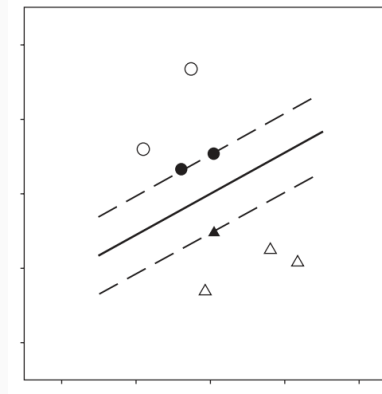
Traducción y Clasificación

Ejemplos de modelos

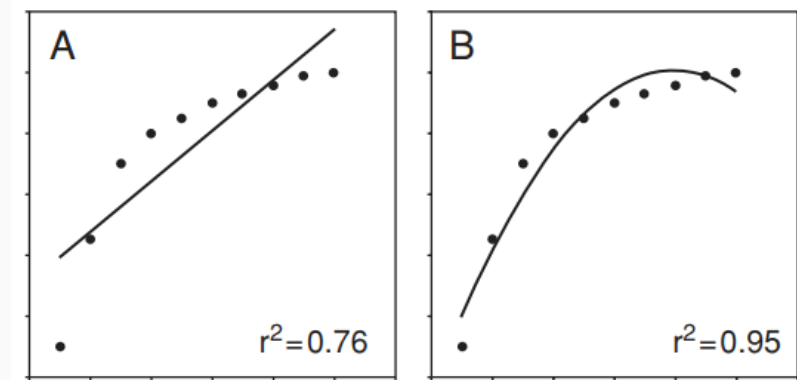
- Discriminantes lineales del tipo $Y = b_1x_1 + b_2x_2 \dots b_nx_n + a$.
- Clasificadores Bayesianos.
- Máquinas de soporte vectorial.
- Modelos no lineales,
 - Regresiones.
 - **Redes neuronales.**
 - Otras



Discriminantes lineales



SVM



Modelo lineal vs no lineal

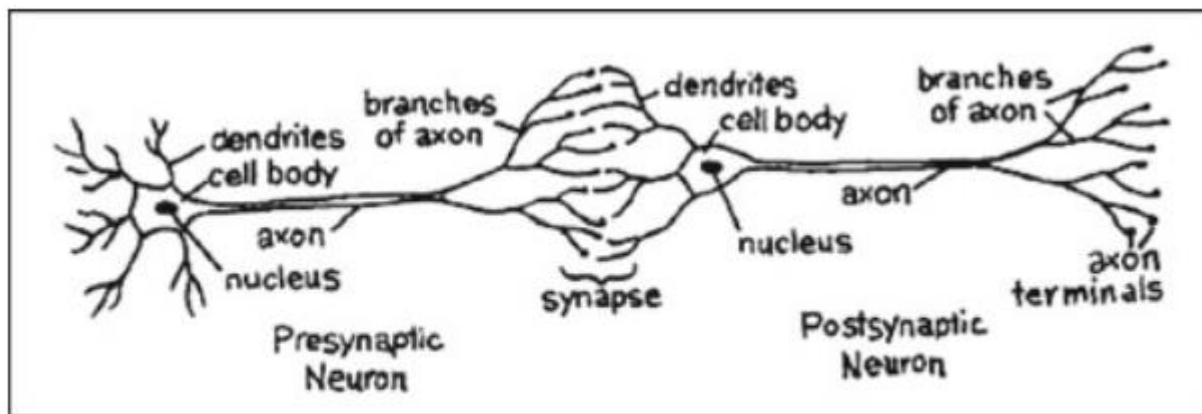
Breaktime!



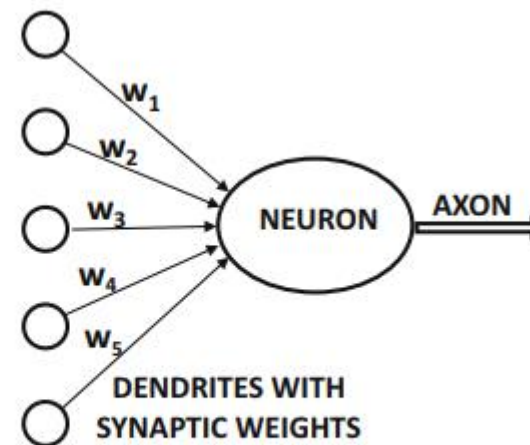
Redes neuronales

Introducción

Las redes neuronales artificiales son algoritmos de machine Learning que simulan la forma en que los organismos aprenden.



(a) Biological neural network



(b) Artificial neural network

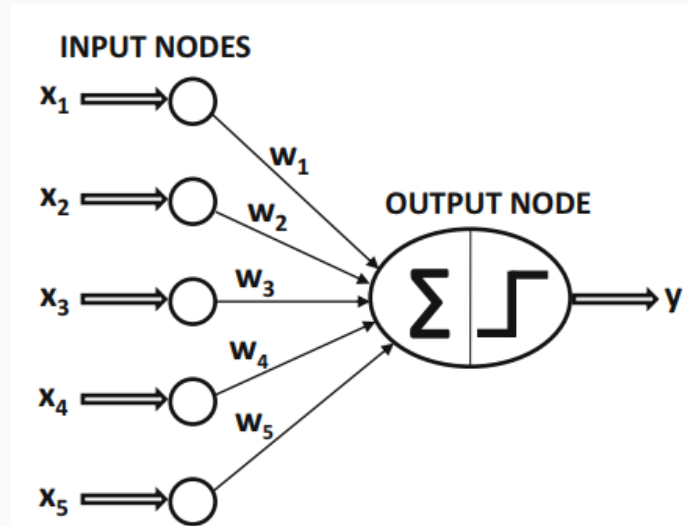
Redes neuronales

Perceptrón

La unidad básica de una NN es el perceptrón.

Un set de parámetros de entradas es mapeado a una salida.

La capa -layer- de entrada contiene d nodos que transmite las d características de mi vector de características $\bar{X} = [x_1 \dots x_d]$ ponderadas con $\bar{W} = [w_1 \dots w_d]$ hacia un nodo de salida.



$$\hat{y} = \text{sign}(\bar{W} \cdot \bar{X}) = \text{sign} \left\{ \sum_{j=1}^d w_j x_j \right\}$$

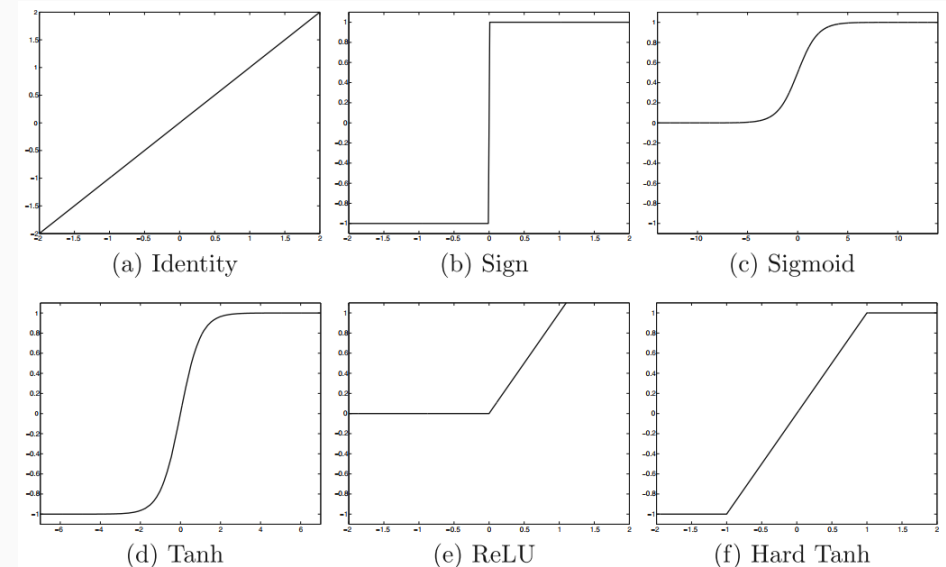
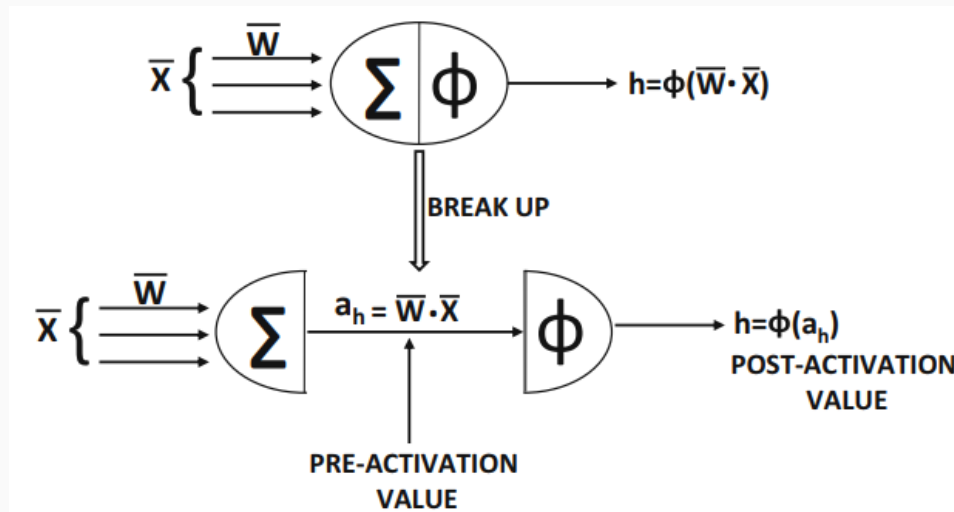
$$\text{Error} = y - \hat{y}$$

Redes neuronales

Función de activación

Las funciones de activación se aplican luego de cada nodo de salida. Sirven similar diferentes tipos modelos usados en machine learning, tales como SVM, clasificadores de regresión logística, entre otros.

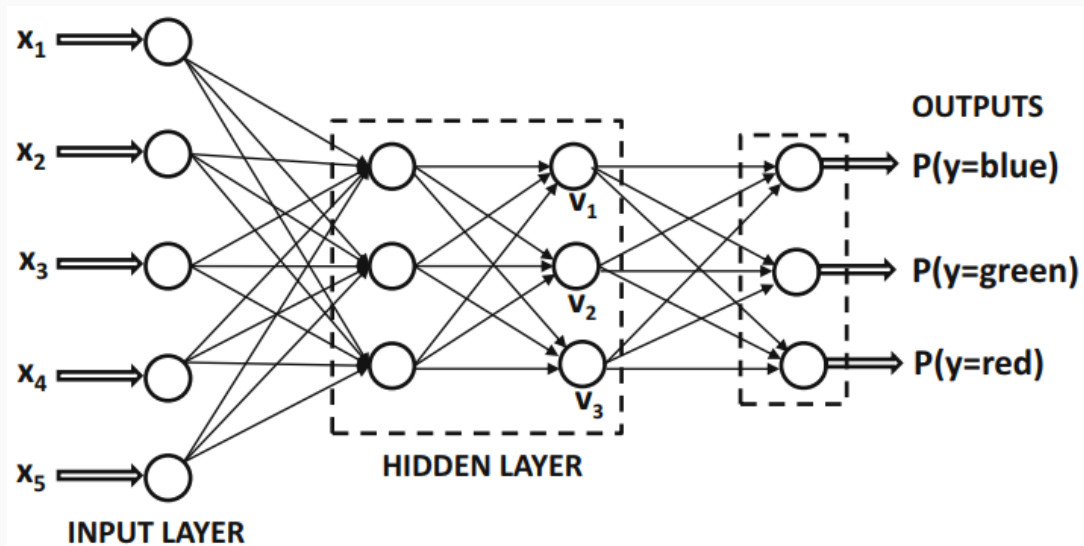
La salida puede estar dada por $\hat{y} = \Phi(\bar{W} \cdot \bar{X})$ donde Φ es la función de activación.



Redes neuronales

Redes multicapa

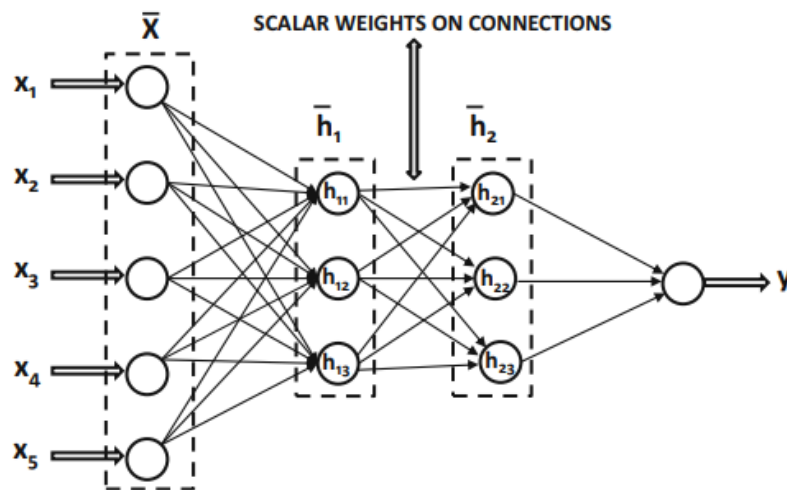
Las NN multicapas poseen más de una layer en su arquitectura. Aquellas capas entre las capas de entrada y salida se conocen cómo *hidden layers*. Este tipo de arquitecturas recibe el nombre de *feed-forward network*.



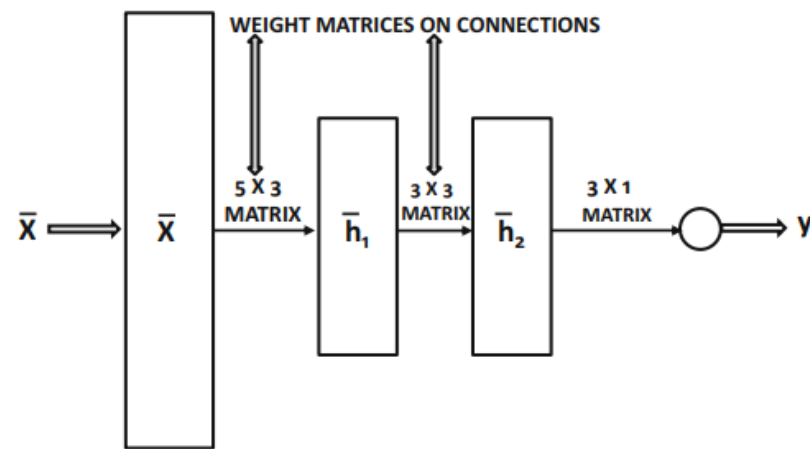
Redes neuronales

Redes multicapa

Las NN multicapas poseen más de una layer en su arquitectura. Aquellas capas entre las capas de entrada y salida se conocen como *hidden layers*. Este tipo de arquitecturas recibe el nombre de *feed-forward network*.



(c) Scalar notation and architecture



(d) Vector notation and architecture

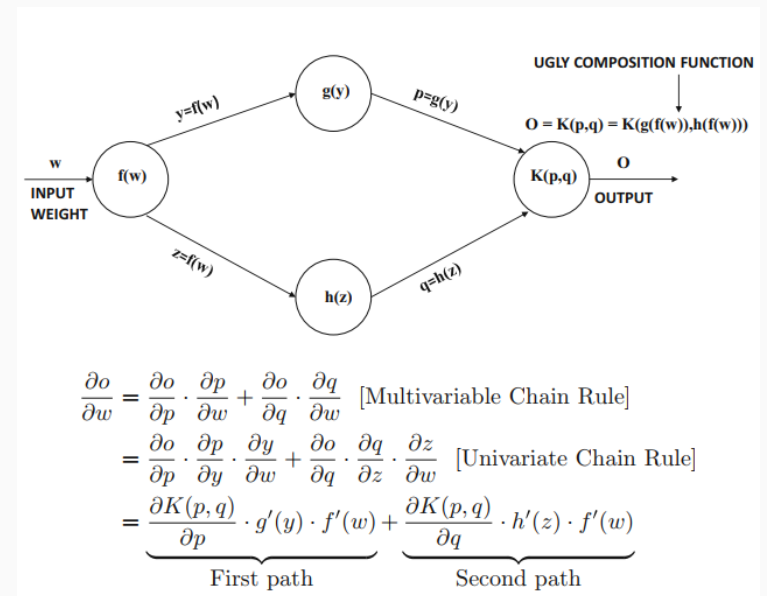
Redes neuronales

Entrenando nuestra red

Para entrenar una red necesitamos datos de entrenamiento.

Con estos datos y conociendo sus salidas, ajustamos los pesos w_{ij} de cada capa de nuestra NN de tal manera de poder clasificar correctamente para cada dato del set de entrenamiento y para futuros datos u observaciones.

Se utiliza el método de *backpropagation*.



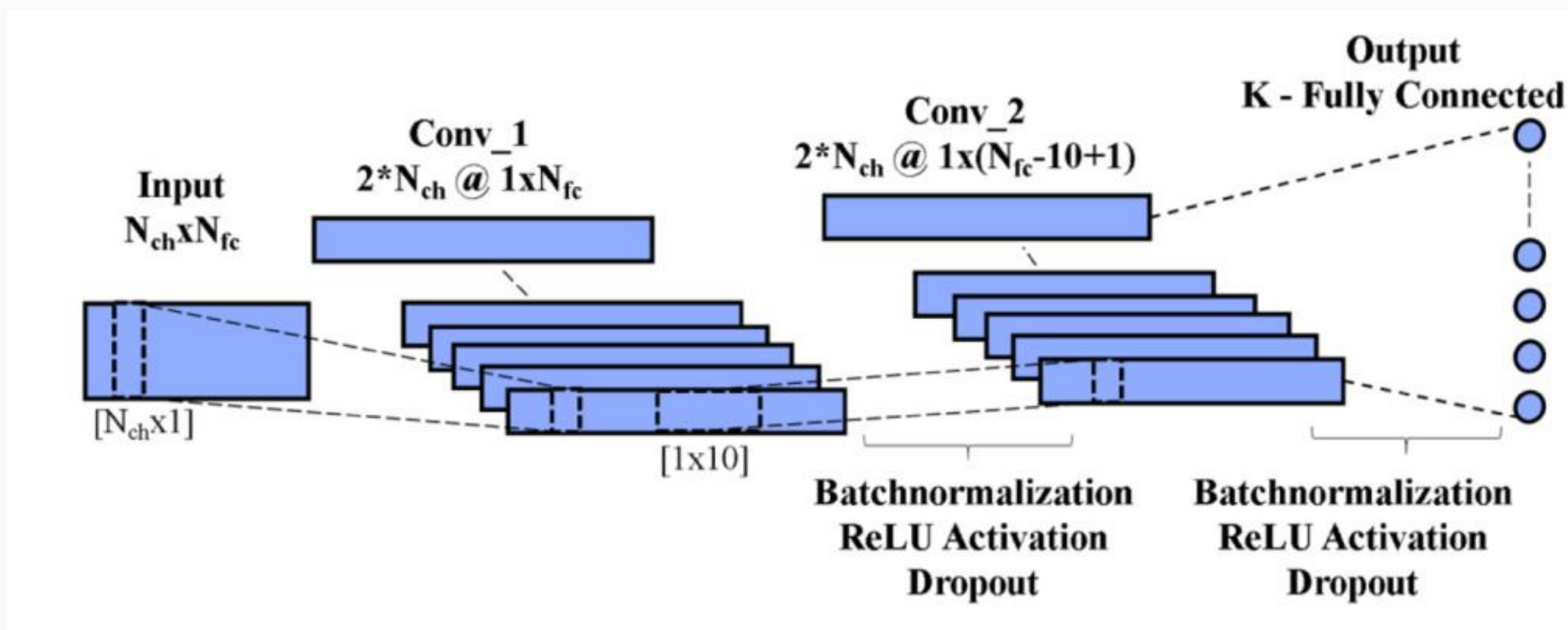
Ejemplo de aplicación



Ejemplo de CNN

Aplicación usando datos reales

Para este ejemplo utilizaremos la arquitectura de la Red Neuronal de Convolución (CNN) propuesta en “*Comparing user-dependent and user-independent training of CNN for SSVEP BCI*”, Aravind Ravi, 2019.

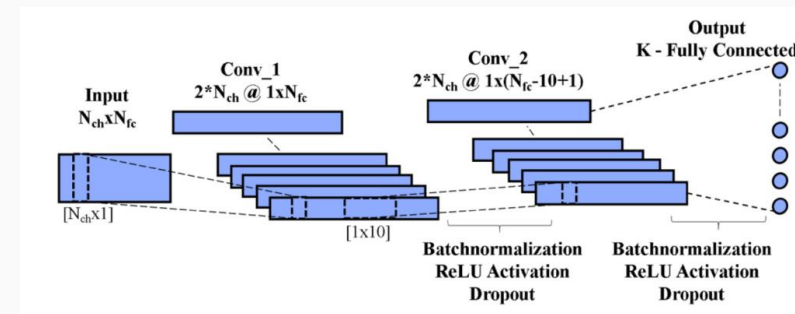


Ejemplo de CNN

Entrenando nuestra CNN

En los scripts compartidos tenemos dos clases importantes,

- **CNNTrainingModule:** Esta clase posee métodos necesarios para entrenar nuestra CNN a partir de señales de EEG.
- **CNNClassify:** Esta clase posee métodos para clasificar a partir de señales de EEG.





Entrenando nuestra CNN

CNNTrainingModule

Pasos para entrenar nuestra CNN.

1. Cargar y pre procesar nuestros datos.
2. Crear nuestro objeto CNNTrainingModule el cual nos permitirá trabajar con los datos del EEG y entrenar nuestra red.
3. Obtener los valores de magnitud del espectro de nuestra señal para usarlo como features.
4. Crear el modelo CNN. Importante: La forma de entrada de los datos -variable inputShape- es fundamental, ya que el modelo creado usando dicha forma, deberá posteriormente recibir los datos exactamente de la misma manera.
5. Quinto: Entrenamos y guardamos el modelo completo con aquellos pesos que produzcan la mejor precisión de clasificación.



Clasificando con nuestra CNN

CNNClassify

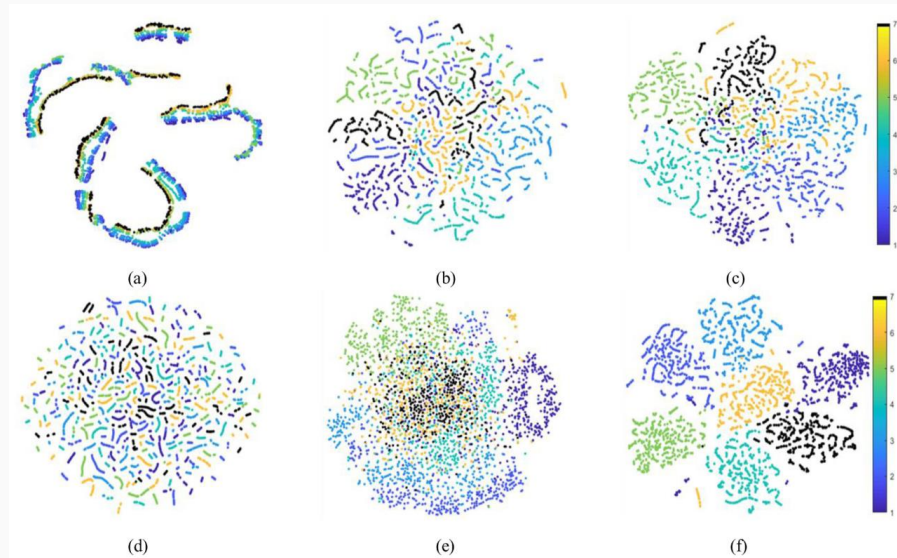
Pasos para usar y clasificar con CNN entrenada.

1. Crear nuestro objeto CNNClassify el cual nos permitirá cargar un modelo previamente entrenado y clasificar datos nuevos.
2. Obtener las características a partir del espectro de Fourier de las señales de EEG que queremos clasificar. Debemos tener en cuenta que para este ejemplo estamos usando un set de datos preexistentes, no obstante, la idea es usar la CNN para clasificar nuevos datos en tiempo real.
3. Utilizar las características obtenidas en el paso dos para clasificar nuestra señal de EEG y así obtener un comando.

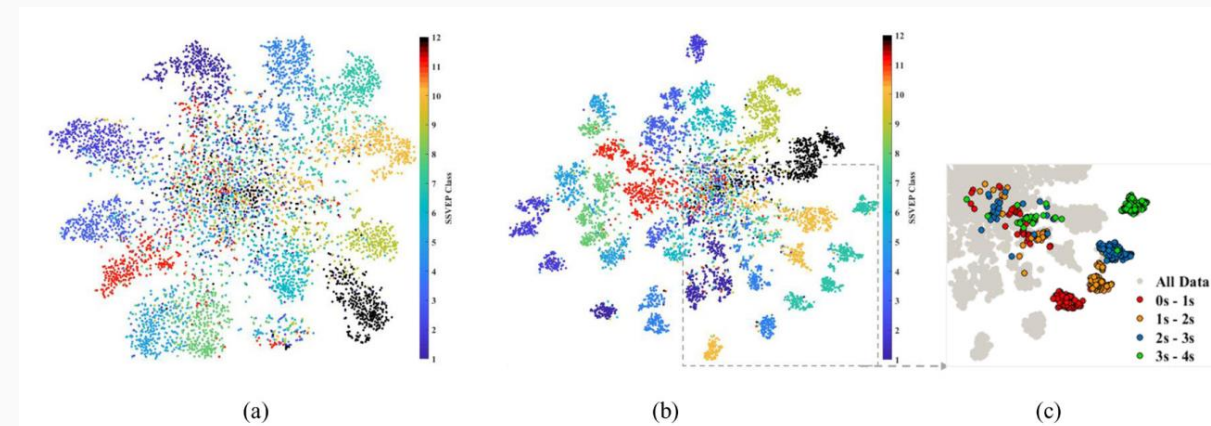
Últimos comentarios

Analizando las características utilizadas para clasificar

Algunos comentarios acerca del método de clasificación a partir de las características extraídas mediante la FFT.

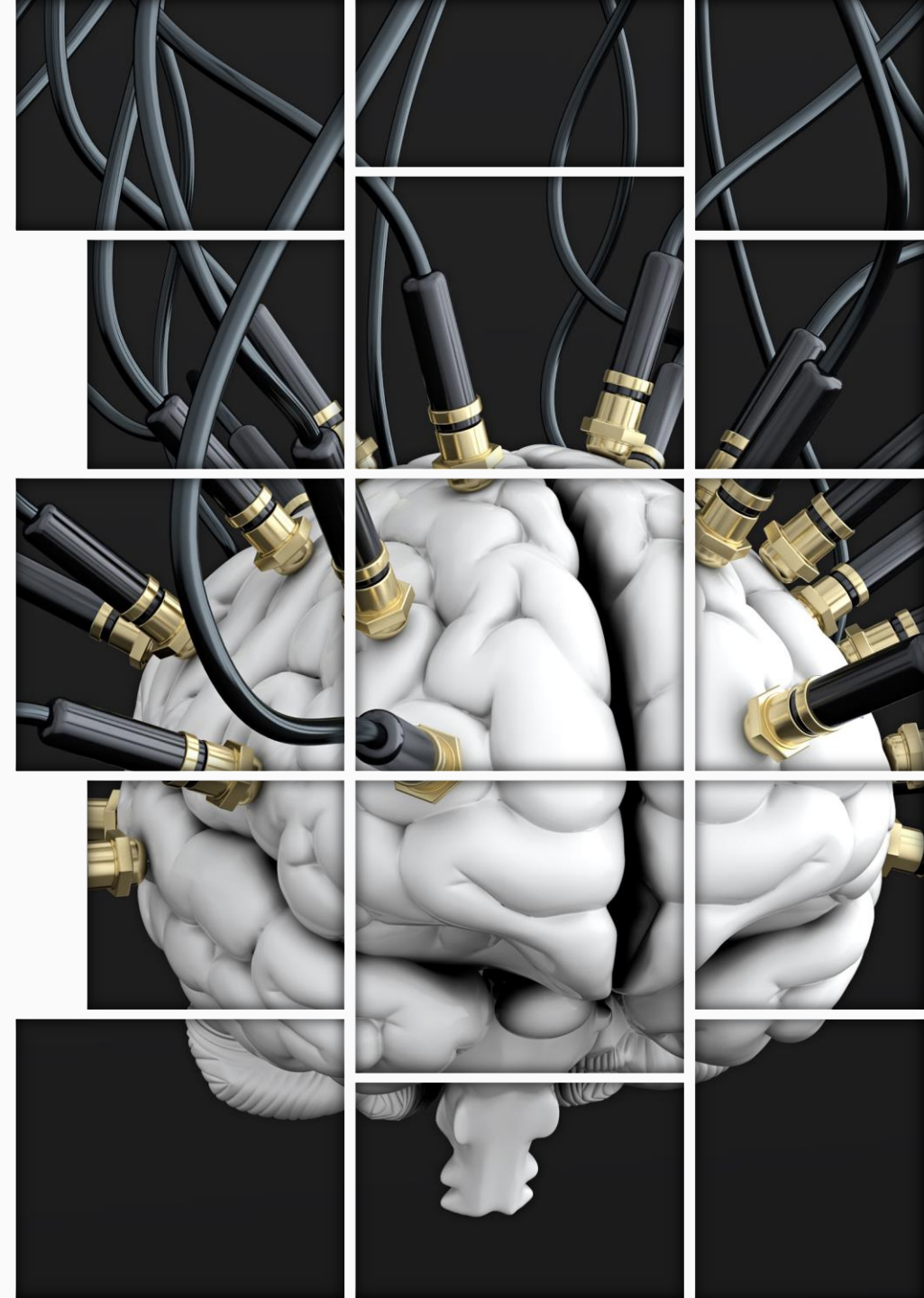


7 clases



12 clases

¿Preguntas?





Bibliografía

[1] *“Brain Computer Interfaces: Principles and Practice”*. Wolpaw Jonathan. Oxford. Published by Oxford University Press, Inc. 2012.

[2] *“Neural Network and Deep Learning”*. Charu C. Aggarwal. Editorial Springer. 2018.

[3] *“Comparing user-dependent and user-independent training of CNN for SSVEP BCI”*. Aravind Ravi. Department of Systems Design Engineering, University of Waterloo, Waterloo, ON, Canada.

Taller N°4

Introducción a la clasificación de señales EEG

Mgr. Bioing. Baldezzari Lucas
Profesor Encargado
Ingeniería Biomédica

