

# Reporte Análisis de datos de actividad cerebral

José Antonio Juárez Pacheco

A00572186

June 14, 2023

# 1 ¿Qué es la electroencefalografía?

La electroencefalografía es un área de la medicina que se especializa en observar el cerebro para poder registrar la actividad bioeléctrica cerebral en donde el paciente se encuentra en reposo o sueño, y también durante diversas activaciones, esto se hace con el fin de poder detectar enfermedades, tales como tumores cerebrales, trastornos del sueño, encefalopatía, entre otros. La actividad cerebral se recolecta con electrodos o potenciales eléctricos que se ponen en el cuero cabelludo del paciente. Así mismo, otros lugares en donde se pueden poner estos potenciales eléctricos es sobre la cabeza, en los lóbulos de las orejas y debajo de los lóbulos.

Así mismo, en la electroencefalografía la actividad cerebral se representa en forma de ondas que se pueden interpretar como una composición de oscilaciones y/o ritmos. Además, existe una clasificación de estas ondas que se llegan a ver ya sea que el sujeto esté en reposo o haciendo actividades cognitivas. La forma en la que se dividen estas ondas es:

- Delta: 0 - 4 Hz
- Theta: 4 - 8 Hz
- Alfa: 8 - 12 Hz
- Beta: 12 - 30 Hz
- Gamma: > 30 Hz

Las ondas Delta y Theta se presentan con mayor frecuencia y la Beta y Gamma no se presentan casi nada cuando el sujeto está en un estado de relajación. Sin embargo, cuando el sujeto está haciendo una tarea cognitiva, los papeles se invierten, tenemos que las ondas Beta y Gamma son las que más frecuencia tienen y la Delta y Theta no se ven tanto.

## 2 Características de las señales que se adquirieron

El experimento realizado se hizo con un equipo llamado **Unicorn Hybrid Black** distribuida por la empresa **gTec**. Así mismo este modelo cuenta con 8 electrodos que están distribuidos de forma estratégica a través de la malla que te pones en tu cabeza para poder agarrar. Esta distribución de los electrodos está basada en el sistema 10-20, el cual habla sobre como tienen que tener un porcentaje de distancia (10 y 20) en la que deben estar colocados los electrodos en la parte frontal, occipital, de lado derecho e izquierda del cráneo.

Otro dato importante de este experimento es que este instrumento tiene una velocidad de muestreo de 250 mediciones por segundo, es decir, 250 Hz. Por lo que al nosotros configurar el experimento a que nada más este abierto por 4 segundos para hacer mediciones, tenemos alrededor de 1000 muestras para poder analizar. Finalmente para terminar de hablar sobre las características de las señales, es importante mencionar que para poder realizar los experimentos, las señales de los 4 electrodos tenían que estar en un rango de oscilación entre -1000 a 1000  $\mu v$ .

### 3 Descripción de los experimentos

- P300

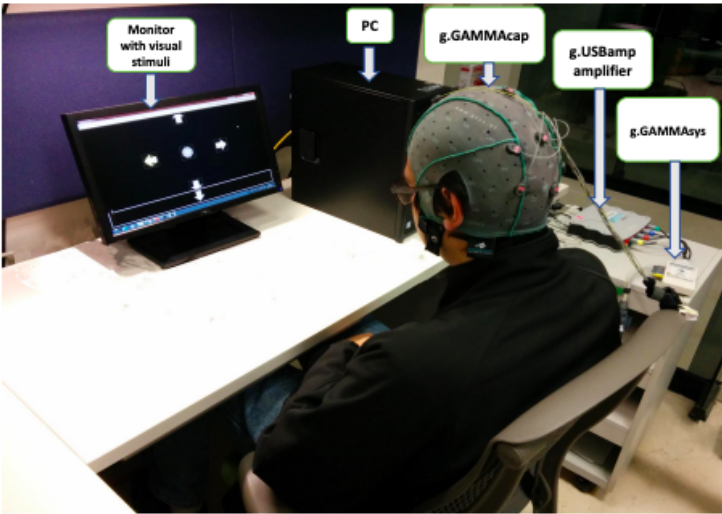


Figure 1:  
En esta imagen podemos ver el material necesario para realizar el experimento. Así mismo, vemos lo que el sujeto necesita para hacer este experimento.

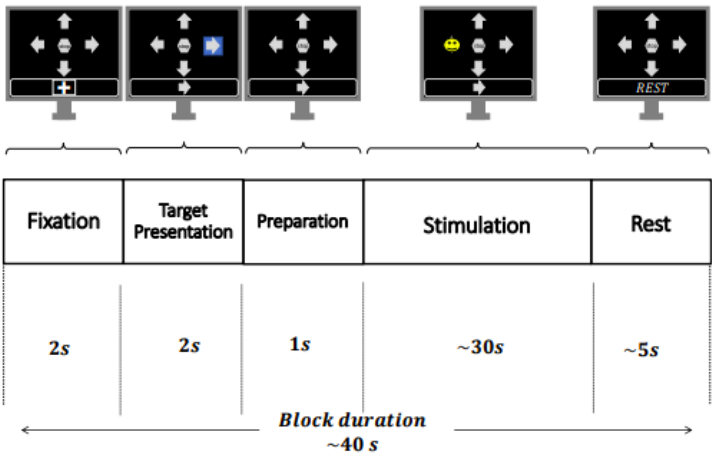


Figure 2:  
En esta imagen podemos ver de una forma muy gráfica el funcionamiento del experimento con los tiempos y lo que hace la simulación

- **Tarea cognitiva**

El experimento de tarea cognitiva se basa en medir las señales del sujeto cuando esté está haciendo una tarea cognitiva vs cuando no se está haciendo una tarea cognitiva. De esta forma se va a poder analizar que frecuencias son las que predenominan en estas dos situaciones. El material y la implementación es la misma a la del experimento P300.

Cuando no se hace una tarea cognitiva, es cuando aparece una cruz en la pantalla y cuando se hace una tarea cognitiva es cuando sale alguna de las 12 pruebas que existen. Estas 12 pruebas están divididas en 3 áreas, una es la de imaginar cosas, otra son operaciones aritméticas y la otra es leer un texto.

A continuación pondré fotos del banco de preguntas para terminar de explicar el experimento



Figure 3: Tarea Cognitiva 1

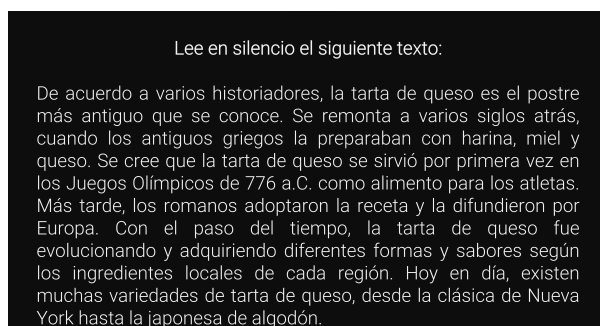


Figure 4: Tarea Cognitiva 2

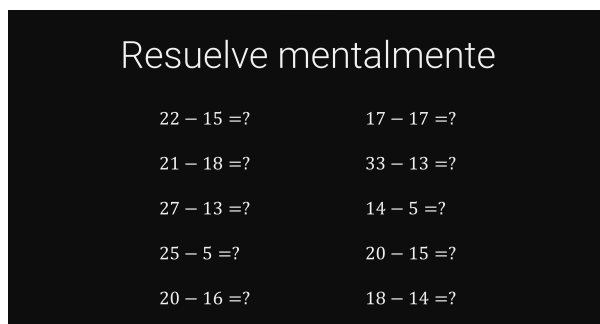


Figure 5: Tarea Cognitiva 3

## 4 Extracción de características

- **P300**

La extracción de características en el experimento P300 se divide en cuatro pasos.

Primer paso: Tenemos 8 canales o señales que se activan cuando aparece la carita feliz en el experimento. Después de eso se extraen 800 milisegundos de muestra, con esto se consiguen 200 muestras por señal, ya que la velocidad de muestreo del experimento es de 250 Hz. De esta forma, lo que se llega a analizar es el conjunto de observaciones que existe en cada uno de los 8 canales con sus respectivas muestras.

Segundo paso: El siguiente paso es hacer un filtrado porque no se necesitan todos los componentes oscilatorios en la actividad P300 que se quiere observar. Así mismo, se hace un filtrado pasabandas, es decir se filtra entre 4 y 20 Hz, de tal manera que se quedan las mismas muestras en las señales pero se eliminan todos los componentes de baja frecuencia y todos los componentes de arriba de 20 Hz.

Tercer paso: El tercer paso es algo que se llama submuestreo. El submuestreo descarta observaciones en señales para así poder analizar menos. Por ejemplo, vamos a submuestrear cada 4 observaciones, la primera observación es la que analizas y las tres siguientes a esta la descartas, y esto es un ciclo que se repite hasta que se acaben las observaciones en la señal, para así nada más quedarte con pocos valores para poder comprimir los datos y analizarlos. Aplicado al experimento, en lugar de tener 200 muestras para cada canal de los ocho que son, aplicamos submuestreo de 4 y ahora por señal tenemos 50 muestras y lo mejor de este proceso es que no se pierde información gracias a las propiedades del filtrado.

El cuarto paso: Este es el paso más importante y es el que hace que funcione la extracción de características en el experimento P300, en este paso se aplica una técnica llamada filtrado espacial. El filtrado espacial combina las señales para así poder obtener nuevas señales, esto se hace de la misma forma en la que se hacen los promedios ponderados. Estas nuevas señales tienen una propiedad que esté mejorada con respecto a las señales originales. Un ejemplo puede ser que la nueva señal tenga menos ruido que las originales. Ahora, al aplicar el filtrado especial en las muestras del experimento P300, se combinan las 8 señales y ahora se quedan simplemente 4 señales nuevas y mejoradas. Así mismo, cada una va a seguir con 50 muestras. Finalmente, estas 4 señales nuevas con 50 muestras cada una, son las características que se utilizan en los modelos de clasificación implementados posteriormente.

- **Tarea cognitiva**

El proceso de extracción de características en la tarea cognitiva funciona de la siguiente manera:

Cuando la cruz aparece en el experimento, este periodo de tiempo se divide en 10 ventanas de 1 segundo. Así mismo, la tarea cognitiva del experimento se divide en 15 ventanas de 1 segundo. Las ventanas, tiene 8 canales o señales con 250 muestras, esto es igual que en el P300. Así mismo, para cada señal se hace un análisis de potencia espectral (PSD).

En el PSD se dice que a mayor energía es mayor amplitud de una oscilación. Entonces lo único importante es analizar cual es el aporte de energía en cada una de las señales, de esta forma se puede analizar cual de las frecuencias explicadas al inicio se frecuentan más dependiendo de si estas realizando una tarea cognitiva o descansado. Finalmente, estas son las características que se van a utilizar en los modelos de clasificación de este reporte.

## 5 Random Forest Classifier

Random Forest Classifier es un algoritmo de aprendizaje inteligente que se basa en el concepto de conjunto de árboles de decisión. Su funcionamiento se basa en la combinación de múltiples árboles de decisión para obtener una clasificación precisa y estable. Ahora, se describirán sus características principales:

- **Construcción de árboles de decisión:** El Random Forest Classifier utiliza árboles de decisión, que son modelos de aprendizaje inteligentes donde se toman decisiones siguiendo una estructura de árbol. Cada nodo representa una pregunta o condición sobre los datos y las ramas representan las posibles respuestas o resultados. Los árboles de decisión se construyen mediante un proceso de aprendizaje supervisado a partir de un conjunto de datos de entrenamiento.
- **Muestreo aleatorio y bootstrap:** Para construir el conjunto de árboles, el algoritmo utiliza un método llamado bootstrap. El bootstrap básicamente se enfoca en generar múltiples conjuntos de datos de entrenamiento mediante muestreo aleatorio con reemplazo.
- **Muestreo aleatorio de características:** En cada nodo de un árbol de decisión, se selecciona un subconjunto aleatorio de variables para así poder tomar una decisión. Esto se conoce como "muestreo aleatorio de características". La selección aleatoria de características ayuda a reducir la dependencia de una sola característica y permite que diferentes árboles tengan diferentes perspectivas de los datos.
- **Votación para la clasificación:** Ya que se construyeron los árboles, se utiliza el proceso de votación para realizar la clasificación. Cuando se recibe una nueva instancia, cada árbol de decisión emite una clasificación. En el caso de la clasificación, la clase más votada se considera la predicción final del método de clasificación Random Forest.
- **Importancia de las características:** Random Forest también trata de enfocar en la importancia de las características. Esta medida se hace al reducir la impureza de los nodos del árbol y se calcula al evaluar cuánto se degrada el rendimiento del modelo cuando se elimina una característica en particular.

Algunas ventajas del método Random Forest Classifier :

- Es altamente eficaz en problemas de clasificación y regresión.
- Puede manejar conjuntos de datos grandes con una alta dimensionalidad.
- Es resistente al sobreajuste debido a la combinación de múltiples árboles y el uso de muestreo aleatorio.
- Proporciona una estimación de la importancia de las características.
- Es capaz de manejar características categóricas y numéricas sin necesidad de preprocesamiento adicional.
- Tiene buen rendimiento incluso en presencia de valores faltantes en los datos.
- Es relativamente rápido en comparación con otros algoritmos de aprendizaje automático.

En conclusión, Random Forest Classifier es un método muy bueno porque utiliza un conjunto de árboles de decisión contruidos a partir de muestras aleatorias y características seleccionadas aleatoriamente para realizar clasificaciones precisas y estables. Además de que tiene una gran capacidad para manejar conjuntos de datos grandes y estimar la importancia de las características

## 6 Resultados

Para este proyecto se obtuvieron 8 bases de datos. Tenemos 2 bases de datos para el experimento de P300. Así mismo, tenemos 3 bases de datos del experimento de tareas cognitivas, en donde, se dividen las diferentes preguntas que existen y se ve como estas diferentes tareas cognitivas afectan la frecuencia. Finalmente, contamos con otras 3 bases de datos del experimento de tareas cognitivas, pero ahora se divide en si se está haciendo una actividad cognitiva o no. Estos resultados, están divididos en diferentes áreas que serán presentadas a continuación:

- **Evaluación de algoritmos de clasificación**

Para este trabajo se utilizaron 5 métodos de clasificación para evaluar las bases de datos de los experimentos P300 y Tareas cognitivas vs no cognitivas. A continuación explicaré los conceptos de estos métodos.

- SVM Classifier: SVM es un clasificador que busca encontrar un hiperplano óptimo para separar los datos en diferentes clases. Busca maximizar el margen entre las clases y utiliza vectores de soporte, que son los puntos de datos más cercanos al hiperplano de separación

- KNN Classifier: KNN clasifica un punto de datos desconocido asignándole la etiqueta de clase más común entre sus vecinos más cercanos en el espacio de características. Los vecinos se seleccionan según una medida de distancia, generalmente la distancia euclidiana.

- MLP Classifier: MLP es una red neuronal artificial que consta de múltiples capas de unidades llamadas neuronas. Las neuronas se organizan en capas de entrada, ocultas y de salida, y utilizan una función de activación para introducir no linealidad. El MLP aprende a través de retropropagación del error y ajusta los pesos de las conexiones para mejorar la precisión de la clasificación.

- Decision Tree Classifier: Un árbol de decisión divide repetidamente el conjunto de datos en subconjuntos más pequeños según las características relevantes. Cada división se basa en una regla o criterio que maximiza la homogeneidad dentro de cada subconjunto y la heterogeneidad entre los subconjuntos.

- Random Forest Classifier: Un Random Forest es un conjunto de árboles de decisión. Cada árbol se entrena con una muestra aleatoria del conjunto de datos y realiza una clasificación individual. La clasificación final se obtiene por votación o promediado de las clasificaciones de los árboles individuales. Así mismo, un mayor número de árboles generalmente reduce el sobreajuste y mejora la precisión, pero también aumenta el costo computacional

- **Clasificación datos P300**

En la clasificación de base de datos P300 como se mencionó son 2 bases de datos. EL primer sujeto se llama Antonio y el segundo sujeto se llama Mayra.

- Datos P300 Antonio:

SVM					
	precision	recall	f1-score	support	
1	0.89	0.56	0.69	57	
2	0.90	0.98	0.94	222	
accuracy			0.90	279	
macro avg	0.89	0.77	0.81	279	
weighted avg	0.90	0.90	0.89	279	
KNN					
	precision	recall	f1-score	support	
1	0.68	0.30	0.41	57	
2	0.84	0.96	0.90	222	
accuracy			0.83	279	
macro avg	0.76	0.63	0.66	279	
weighted avg	0.81	0.83	0.80	279	
MLP					
	precision	recall	f1-score	support	
1	0.75	0.59	0.66	56	
2	0.90	0.95	0.93	223	
accuracy			0.88	279	
macro avg	0.83	0.77	0.79	279	
weighted avg	0.87	0.88	0.87	279	

Figure 6: SVM, KNN, MLP. Resultados son accuracy f1-score.

Decision Tree					
	precision	recall	f1-score	support	
1	0.43	0.46	0.44	57	
2	0.86	0.85	0.85	222	
accuracy			0.77	279	
macro avg	0.65	0.65	0.65	279	
weighted avg	0.77	0.77	0.77	279	
Random Forest					
	precision	recall	f1-score	support	
1	0.94	0.26	0.41	57	
2	0.84	1.00	0.91	222	
accuracy			0.85	279	
macro avg	0.89	0.63	0.66	279	
weighted avg	0.86	0.85	0.81	279	

Figure 7: Decision tree y Random Forest. Resultados son accuracy f1-score.

En estas tablas podemos ver que el método de clasificación que mejor accuracy tuvo en f1-score fue el método SVM, después MLP, Random Forest, KNN y finalmente Decision Tree. Esto quiere decir que el método más preciso para esta base de datos fue el Support Vector Machine (SVM).



- Datos P300 Mayra:

SVM					
	precision	recall	f1-score	support	
1	1.00	0.27	0.42		26
2	0.78	1.00	0.87		66
accuracy			0.79		92
macro avg	0.89	0.63	0.65		92
weighted avg	0.84	0.79	0.75		92
KNN					
	precision	recall	f1-score	support	
1	0.67	0.08	0.14		26
2	0.73	0.98	0.84		66
accuracy			0.73		92
macro avg	0.70	0.53	0.49		92
weighted avg	0.71	0.73	0.64		92
MLP					
	precision	recall	f1-score	support	
1	0.50	0.21	0.30		19
2	0.82	0.95	0.88		73
accuracy			0.79		92
macro avg	0.66	0.58	0.59		92
weighted avg	0.76	0.79	0.76		92

Figure 8: SVM, KNN, MLP. Resultados son accuracy f1-score.

Decision Tree					
	precision	recall	f1-score	support	
1	0.45	0.35	0.39		26
2	0.76	0.83	0.80		66
accuracy			0.70		92
macro avg	0.61	0.59	0.59		92
weighted avg	0.68	0.70	0.68		92
Random Forest					
	precision	recall	f1-score	support	
1	1.00	0.12	0.21		26
2	0.74	1.00	0.85		66
accuracy			0.75		92
macro avg	0.87	0.56	0.53		92
weighted avg	0.81	0.75	0.67		92

Figure 9: Decision tree y Random Forest. Resultados son accuracy f1-score.

En estas tablas, al igual que con la base de datos P300 de Antonio, podemos ver que el método de clasificación que mejor accuracy tuvo en f1-score fue el método SVM, después MLP, Random Forest, KNN y finalmente Decision Tree. Esto quiere decir que el método más preciso para esta base de datos fue el Support Vector Machine (SVM). De esta forma podemos ver un patrón que nos da a entender que para todas las bases de datos de este estilo el mejor método de clasificación siempre va a ser SVM.

- **Clasificación de datos cognitivo**

Para esta clasificación contamos con 3 bases de datos que esta dividida en 4 clases dependiendo en la tarea cognitiva que se va a realizar. La primer base de dato es de un sujeto llamado Antonio, la segunda de un sujeto llamado Mayra y la tercera de un sujeto llamado Raúl. Para cada una de estas bases de datos vamos a aplicar los 5 métodos de clasificación.

- Datos Cognitivo Antonio:

SVM					
	precision	recall	f1-score	support	
1	0.40	0.17	0.24	12	
2	0.33	0.22	0.27	9	
3	0.75	0.19	0.30	16	
4	0.41	0.89	0.57	19	
accuracy			0.43	56	
macro avg	0.47	0.37	0.34	56	
weighted avg	0.49	0.43	0.37	56	
KNN					
	precision	recall	f1-score	support	
1	0.20	0.17	0.18	12	
2	0.30	0.67	0.41	9	
3	0.44	0.25	0.32	16	
4	0.65	0.58	0.61	19	
accuracy			0.41	56	
macro avg	0.40	0.42	0.38	56	
weighted avg	0.44	0.41	0.40	56	
MLP					
	precision	recall	f1-score	support	
1	1.00	0.00	0.00	12	
2	1.00	0.00	0.00	11	
3	1.00	0.00	0.00	11	
4	0.39	1.00	0.56	22	
accuracy			0.39	56	
macro avg	0.85	0.25	0.14	56	
weighted avg	0.76	0.39	0.22	56	

Figure 10: SVM, KNN, MLP. Resultados son accuracy f1-score.

Decision Tree					
	precision	recall	f1-score	support	
1	0.17	0.17	0.17	12	
2	0.38	0.56	0.45	9	
3	0.50	0.44	0.47	16	
4	0.47	0.42	0.44	19	
accuracy			0.39	56	
macro avg	0.38	0.40	0.38	56	
weighted avg	0.40	0.39	0.39	56	
Random Forest					
/usr/local/lib/python3.10/dist-packages/sklearn/metrics					
_warn_prf(average, modifier, msg_start, len(result))					
/usr/local/lib/python3.10/dist-packages/sklearn/metrics					
_warn_prf(average, modifier, msg_start, len(result))					
/usr/local/lib/python3.10/dist-packages/sklearn/metrics					
_warn_prf(average, modifier, msg_start, len(result))					
	precision	recall	f1-score	support	
1	0.27	0.25	0.26	12	
2	0.44	0.78	0.56	9	
3	0.50	0.06	0.11	16	
4	0.56	0.79	0.65	19	
accuracy			0.46	56	
macro avg	0.44	0.47	0.40	56	
weighted avg	0.46	0.46	0.40	56	

Figure 11: Decision tree y Random Forest. Resultados son accuracy f1-score.

En estas tablas podemos ver que el método de clasificación que mejor accuracy tuvo en f1-score fue el método Random Forest, SVM, después KNN, MLP y finalmente Decision Tree. Esto quiere decir que el método más preciso para esta base de datos fue el Random Forest Classifier (RFC).

- Datos Cognitivo Mayra:

SVM					
	precision	recall	f1-score	support	
1	1.00	0.00	0.00	11	
2	0.82	0.90	0.86	10	
3	1.00	0.15	0.27	13	
4	0.48	0.95	0.63	21	
accuracy			0.56	55	
macro avg	0.82	0.50	0.44	55	
weighted avg	0.77	0.56	0.46	55	
KNN					
	precision	recall	f1-score	support	
1	0.00	0.00	0.00	11	
2	0.75	0.90	0.82	10	
3	0.71	0.38	0.50	13	
4	0.51	0.86	0.64	21	
accuracy			0.58	55	
macro avg	0.49	0.54	0.49	55	
weighted avg	0.50	0.58	0.51	55	
MLP					
	precision	recall	f1-score	support	
1	1.00	0.00	0.00	12	
2	0.58	0.64	0.61	11	
3	0.00	0.00	0.00	11	
4	0.50	1.00	0.67	21	
accuracy			0.51	55	
macro avg	0.52	0.41	0.32	55	
weighted avg	0.53	0.51	0.38	55	

Figure 12: SVM, KNN, MLP. Resultados son accuracy f1-score.

Decision Tree					
/usr/local/lib/python3.10/dist-packages/sklearn/metrics					
_warn_prf(average, modifier, msg_start, len(result))					
/usr/local/lib/python3.10/dist-packages/sklearn/metrics					
_warn_prf(average, modifier, msg_start, len(result))					
/usr/local/lib/python3.10/dist-packages/sklearn/metrics					
_warn_prf(average, modifier, msg_start, len(result))					
	precision	recall	f1-score	support	
1	0.25	0.36	0.30	11	
2	0.64	0.70	0.67	10	
3	0.60	0.46	0.52	13	
4	0.72	0.62	0.67	21	
accuracy			0.55	55	
macro avg	0.55	0.54	0.54	55	
weighted avg	0.58	0.55	0.56	55	
Random Forest					
	precision	recall	f1-score	support	
1	0.60	0.27	0.37	11	
2	0.90	0.90	0.90	10	
3	0.71	0.38	0.50	13	
4	0.55	0.86	0.67	21	
accuracy			0.64	55	
macro avg	0.69	0.60	0.61	55	
weighted avg	0.66	0.64	0.61	55	

Figure 13: Decision tree y Random Forest. Resultados son accuracy f1-score.

En estas tablas podemos ver que el método de clasificación que mejor accuracy tuvo en f1-score fue el método Random Forest, KNN, después SVM, Decision Tree y finalmente MLP. Esto quiere decir que el método más preciso para esta base de datos fue el Random Forest Classifier (RFC).

- Datos Cognitivo Raúl:

SVM					
	precision	recall	f1-score	support	
1	1.00	0.00	0.00	11	
2	1.00	0.00	0.00	10	
3	1.00	0.00	0.00	13	
4	0.38	1.00	0.55	21	
accuracy			0.38	55	
macro avg	0.85	0.25	0.14	55	
weighted avg	0.76	0.38	0.21	55	
KNN					
	precision	recall	f1-score	support	
1	0.70	0.64	0.67	11	
2	1.00	0.90	0.95	10	
3	0.75	0.23	0.35	13	
4	0.59	0.90	0.72	21	
accuracy			0.69	55	
macro avg	0.76	0.67	0.67	55	
weighted avg	0.73	0.69	0.66	55	
MLP					
	precision	recall	f1-score	support	
1	1.00	0.00	0.00	12	
2	1.00	0.00	0.00	11	
3	0.00	0.00	0.00	11	
4	0.36	0.90	0.51	21	
accuracy			0.35	55	
macro avg	0.59	0.23	0.13	55	
weighted avg	0.56	0.35	0.20	55	

Figure 14: SVM, KNN, MLP. Resultados son accuracy f1-score.

Decision Tree					
	precision	recall	f1-score	support	
1	0.30	0.27	0.29	11	
2	0.71	0.50	0.59	10	
3	0.43	0.46	0.44	13	
4	0.58	0.67	0.62	21	
accuracy			0.51	55	
macro avg	0.51	0.48	0.49	55	
weighted avg	0.51	0.51	0.51	55	
Random Forest					
	precision	recall	f1-score	support	
1	0.82	0.82	0.82	11	
2	0.91	1.00	0.95	10	
3	0.50	0.31	0.38	13	
4	0.68	0.81	0.74	21	
accuracy			0.73	55	
macro avg	0.73	0.73	0.72	55	
weighted avg	0.71	0.73	0.71	55	

Figure 15: Decision tree y Random Forest. Resultados son accuracy f1-score.

En estas tablas podemos ver que el método de clasificación que mejor accuracy tuvo en f1-score fue el método Random Forest, KNN, después Decision Tree, SVM y finalmente MLP. Esto quiere decir que el método más preciso para esta base de datos fue el Random Forest Classifier (RFC). En este experimento el mejor método de clasificación es Random Forest Classifier comparado al experimento de P300 que fue SVM. Esto puede ser por las características de las bases de datos.

- **Clasificación de datos cognitivo vs no cognitivo**

Para esta clasificación contamos con 3 bases de datos que esta dividida en 2 clases dependiendo si se está realizando una tarea cognitiva o no. La primer base de datos es de un sujeto llamado Antonio, la segunda de un sujeto llamado Mayra y la tercera de un sujeto llamado Raúl. Para cada una de estas bases de datos vamos a aplicar los 5 métodos de clasificación.

- Datos cognitivos vs no cognitivos Antonio:

SVM					
	precision	recall	f1-score	support	
1	0.79	0.92	0.85	37	
4	0.77	0.53	0.62	19	
accuracy			0.79	56	
macro avg	0.78	0.72	0.74	56	
weighted avg	0.78	0.79	0.77	56	
KNN					
	precision	recall	f1-score	support	
1	0.75	0.97	0.85	37	
4	0.88	0.37	0.52	19	
accuracy			0.77	56	
macro avg	0.81	0.67	0.68	56	
weighted avg	0.79	0.77	0.74	56	
MLP					
	precision	recall	f1-score	support	
1	0.78	0.82	0.80	34	
4	0.70	0.64	0.67	22	
accuracy			0.75	56	
macro avg	0.74	0.73	0.73	56	
weighted avg	0.75	0.75	0.75	56	

Figure 16: SVM, KNN, MLP. Resultados son accuracy f1-score.

Decision Tree					
	precision	recall	f1-score	support	
1	0.84	0.84	0.84	37	
4	0.68	0.68	0.68	19	
accuracy			0.79	56	
macro avg	0.76	0.76	0.76	56	
weighted avg	0.79	0.79	0.79	56	
Random Forest					
	precision	recall	f1-score	support	
1	0.81	0.92	0.86	37	
4	0.79	0.58	0.67	19	
accuracy			0.80	56	
macro avg	0.80	0.75	0.76	56	
weighted avg	0.80	0.80	0.79	56	

Figure 17: Decision tree y Random Forest. Resultados son accuracy f1-score.

En estas tablas podemos ver que el método de clasificación que mejor accuracy tuvo en f1-score fue el método Random Forest, SVM, después Decision Tree, KNN y finalmente MLP. Esto quiere decir que el método más preciso para esta base de datos fue el Random Forest Classifier (RFC).

- Datos cognitivos vs no cognitivos Mayra:

SVM					
	precision	recall	f1-score	support	
1	0.70	0.88	0.78		34
4	0.67	0.38	0.48		21
accuracy			0.69		55
macro avg	0.68	0.63	0.63		55
weighted avg	0.69	0.69	0.67		55
KNN					
	precision	recall	f1-score	support	
1	0.77	0.79	0.78		34
4	0.65	0.62	0.63		21
accuracy			0.73		55
macro avg	0.71	0.71	0.71		55
weighted avg	0.73	0.73	0.73		55
MLP					
	precision	recall	f1-score	support	
1	0.84	0.76	0.80		34
4	0.67	0.76	0.71		21
accuracy			0.76		55
macro avg	0.75	0.76	0.76		55
weighted avg	0.77	0.76	0.77		55

Figure 18: SVM, KNN, MLP. Resultados son accuracy f1-score.

Decision Tree					
	precision	recall	f1-score	support	
1	0.71	0.71	0.71		34
4	0.52	0.52	0.52		21
accuracy			0.64		55
macro avg	0.61	0.61	0.61		55
weighted avg	0.64	0.64	0.64		55
Random Forest					
	precision	recall	f1-score	support	
1	0.82	0.82	0.82		34
4	0.71	0.71	0.71		21
accuracy			0.78		55
macro avg	0.77	0.77	0.77		55
weighted avg	0.78	0.78	0.78		55

Figure 19: Decision tree y Random Forest. Resultados son accuracy f1-score.

En estas tablas podemos ver que el método de clasificación que mejor accuracy tuvo en f1-score fue el método Random Forest, MLP, después KNN, SVM y finalmente Decision Tree. Esto quiere decir que el método más preciso para esta base de datos fue el Random Forest Classifier (RFC).

- Datos cognitivos vs no cognitivos Raúl:

SVM					
	precision	recall	f1-score	support	
1	0.62	1.00	0.76	34	
4	1.00	0.00	0.00	21	
accuracy			0.62	55	
macro avg	0.81	0.50	0.38	55	
weighted avg	0.76	0.62	0.47	55	
KNN					
	precision	recall	f1-score	support	
1	0.82	0.68	0.74	34	
4	0.59	0.76	0.67	21	
accuracy			0.71	55	
macro avg	0.71	0.72	0.70	55	
weighted avg	0.73	0.71	0.71	55	
MLP					
	precision	recall	f1-score	support	
1	1.00	0.00	0.00	34	
4	0.38	1.00	0.55	21	
accuracy			0.38	55	
macro avg	0.69	0.50	0.28	55	
weighted avg	0.76	0.38	0.21	55	

Figure 20: SVM, KNN, MLP. Resultados son accuracy f1-score.

Decision Tree					
	precision	recall	f1-score	support	
1	0.76	0.65	0.70	34	
4	0.54	0.67	0.60	21	
accuracy			0.65	55	
macro avg	0.65	0.66	0.65	55	
weighted avg	0.67	0.65	0.66	55	
Random Forest					
	precision	recall	f1-score	support	
1	0.80	0.82	0.81	34	
4	0.70	0.67	0.68	21	
accuracy			0.76	55	
macro avg	0.75	0.75	0.75	55	
weighted avg	0.76	0.76	0.76	55	

Figure 21: Decision tree y Random Forest. Resultados son accuracy f1-score.

En estas tablas podemos ver que el método de clasificación que mejor accuracy tuvo en f1-score fue el método Random Forest, KNN, después Decision Tree, SVM y finalmente MLP. Esto quiere decir que el método más preciso para esta base de datos fue el Random Forest Classifier (RFC). En este experimento, al igual que en el cognitivo, el mejor método de clasificación es Random Forest Classifier comparado al experimento de P300 que fue SVM. Esto puede ser por las características de las bases de datos.

- **Modelos de clasificación**

Para cada una de las 8 bases de datos, se creó un modelo de clasificación tipo filter. De esta forma podemos ver cual es la cantidad de características suficiente para obtener buenos resultados.

- P300 Antonio:

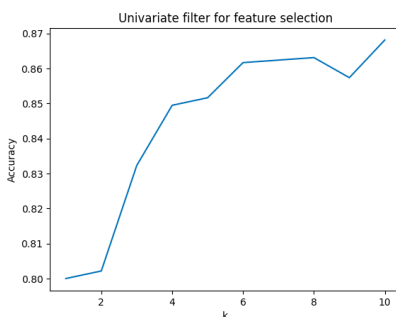


Figure 22: El mejor coeficiente son 10 características. Por ende, esta son las características suficientes para obtener buenos resultados.

- P300 Mayra:

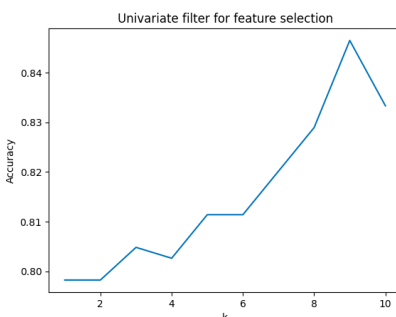


Figure 23: El mejor coeficiente son 9 características. Por ende, esta son las características suficientes para obtener buenos resultados.

- Datos cognitivos Antonio:

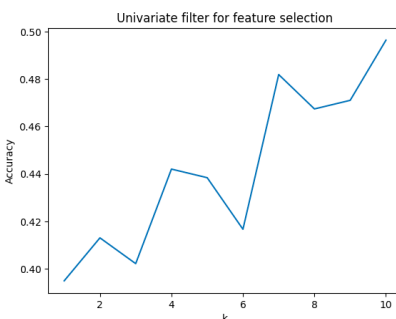


Figure 24: El mejor coeficiente son 10 características. Por ende, esta son las características suficientes para obtener buenos resultados.



- Datos cognitivos Mayra:

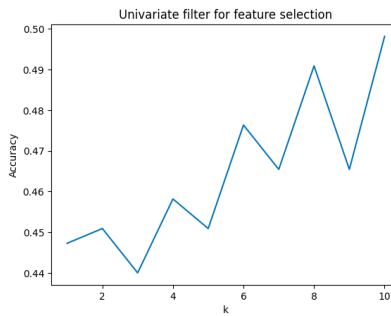


Figure 25: El mejor coeficiente son 10 características. Por ende, esta son las características suficientes para obtener buenos resultados.

- Datos cognitivos Raúl:

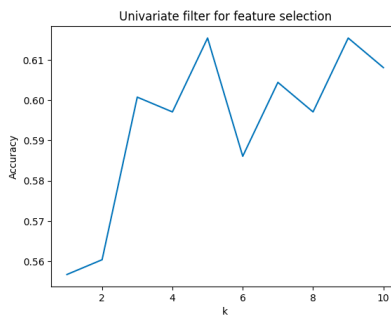


Figure 26: El mejor coeficiente son 5 o 9 características. Por ende, esta son las características suficientes para obtener buenos resultados.

- Datos cognitivos vs no cognitivos Antonio:

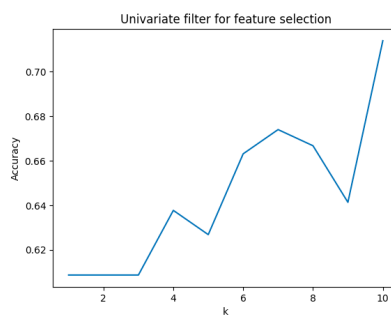


Figure 27: El mejor coeficiente son 10 características. Por ende, esta son las características suficientes para obtener buenos resultados.

- Datos cognitivos vs no cognitivos Mayra:

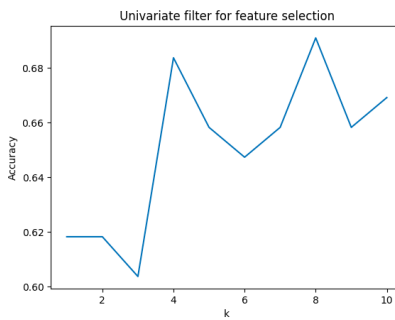


Figure 28: El mejor coeficiente son 8 características. Por ende, esta son las características suficientes para obtener buenos resultados.

- Datos cognitivos vs no cognitivos Raúl:

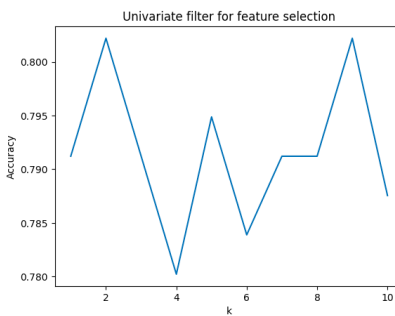


Figure 29: El mejor coeficiente son 2 o 9 características. Por ende, esta son las características suficientes para obtener buenos resultados.

Con este modelo de clasificación podemos observar que cada base de dato necesita un diferente número de características suficientes para obtener buenos resultados.

- **Aprendizaje por transferencia**

En esta última parte del trabajo, se usa las dos bases de datos del experimento P300. Se utilizará una base de dato para entrenar un modelo de clasificación y se probará este modelo con la otra base de datos, de esta forma podremos ver por medio de la precisión si se puede obtener buenos resultados entrenando el modelo con una base de datos y probando el modelo con otra base de datos. Para este aprendizaje se usarán 3 diferentes modelos de clasificación. Finalmente, la base de datos que se usará de entrenamiento será la del sujeto llamado Antonio y la base de datos que se usará de prueba será la del sujeto llamado Mayra. A continuación, se pondrá los resultados de este aprendizaje por transferencia.

- Clasificador SVM:

	precision	recall	f1-score	support
1	0.08	0.02	0.03	92
2	0.79	0.93	0.86	364
accuracy			0.75	456
macro avg	0.43	0.48	0.45	456
weighted avg	0.65	0.75	0.69	456

Figure 30: El accuracy f1-score es 0.75

- Clasificador KNN:

	precision	recall	f1-score	support
1	0.11	0.02	0.04	92
2	0.79	0.95	0.87	364
accuracy			0.77	456
macro avg	0.45	0.49	0.45	456
weighted avg	0.66	0.77	0.70	456

Figure 31: El accuracy f1-score es 0.77

- Clasificador Random Forest Classifier:

	precision	recall	f1-score	support
1	0.07	0.01	0.02	92
2	0.79	0.96	0.87	364
accuracy			0.77	456
macro avg	0.43	0.49	0.44	456
weighted avg	0.65	0.77	0.70	456

Figure 32: El accuracy f1-score es 0.77

Como podemos observar con los modelos de clasificación, los accuracy f1-score son bajos. El que el accuracy sea bajo puede influir que las bases de datos no estén balanceadas, al igual que influye mucho las diferencias que tengan las características y datos de los sujetos. Sin embargo, en un caso en donde las bases de datos estén balanceadas y los sujetos no tengan datos tan dispersos se podría obtener buenos resultados entrenando el modelo con una base de datos y probando el modelo con otra base de datos.

## 7 Conclusión

En este trabajo se presenta como se utiliza la inteligencia artificial para clasificar bases de datos que ya tienen extraídas características importantes de las señales generadas en la cabeza, por medio de instrumento médico que nos permite por medio de sensores, monitorear la actividad cerebral.

El monitoreo de la actividad cerebral es muy beneficioso para la salud de una persona porque mejora la salud mental y el bienestar. Así mismo, diagnostica y trata trastornos cerebrales, entrena y mejora el área cognitiva, hasta se puede aplicar para optimizar el rendimiento deportivo. Es por eso que el monitoreo de la actividad cerebral es un área que tiene demasiados beneficios por las diferentes áreas que mejorar, pero es esencial abordar las consideraciones éticas y de privacidad asociadas con el uso de esta tecnología porque no se quiere romper con los principios de privacidad del usuario.

Además de monitorear la actividad cerebral, otras aplicaciones que llegan a ser muy beneficiosas para la salud y que se pueden monitorear con datos médicos son el monitoreo de la frecuencia cardíaca, el monitoreo de la presión y monitoreo de la glucosa en la sangre. Así mismo, tiene muchos beneficios en la salud monitorear el sueño, la actividad física, la temperatura corporal y la respiración.

Finalmente, lo que más me gusta de estos dispositivos es que tienen la habilidad de poder monitorear cosas de la vida cotidiana que no tienen que ver con el cuerpo humano para poder desarrollar aplicaciones médicas. Por ejemplo, se puede monitorear la calidad del aire y de la contaminación para ver como este puede afectar a los pulmones de las personas. Así mismo, el monitoreo del ritmo circadiano y exposición a la luz puede ayudar a entender y mejorar los patrones de sueño. Otro ejemplo puede ser el monitoreo de la ingesta de nutrientes y alimentos para prevenir enfermedades estomacales y que la persona tenga una vida saludable.

Podemos concluir de esta forma que el monitoreo de datos médicos juega un papel importante en el desarrollo de la medicina porque tiene un campo de trabajo sumamente amplio y muchas aplicaciones que pueden beneficiar al ser humano mediante la innovación de esta área.

## 8 Referencias bibliográficas

- Earningtransfer. (2021, 1 septiembre). Transferencia de aprendizaje: ¿Qué es y por qué es importante? SchoolRubric. SchoolRubric.  
<https://schoolrubric.com/es/transferencia-de-aprendizaje-que-es-y-por-que-es-importante/>
- Electroencefalografía (EEG) - Mayo Clinic. (2022, 19 julio). <https://www.mayoclinic.org/es-es/tests-procedures/eeg/about/pac-20393875>
- Electroencefalograma — ¿Qué es un electroencefalograma — PortalCLÍNICA. (s.f.). Clínic Barcelona.  
<https://www.clinicbarcelona.org/asistencia/pruebas-y-procedimientos/electroencefalograma>
- Heras, J. M. (2020, 18 septiembre). Random Forest (Bosque Aleatorio): combinando árboles - IArtificial.net. IArtificial.net. <https://www.iartificial.net/random-forest-bosque-aleatorio/>
- R, S. E. (2023b). Understand Random Forest Algorithms With Examples (Updated 2023). Analytics Vidhya. <https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/>