

## 22.46 Procesamiento Adaptativo de Señales Aleatorias

# Proyecto 1

### LMS

En este proyecto deberán resolver un problema con el algoritmo LMS, o con enfoques y algoritmos basados en LMS.

Deben **justificar** todas las decisiones que tomen: enfoque al problema, algoritmos, meta-parámetros, criterios de calidad (velocidad de convergencia, desajuste, curva de aprendizaje, bias de coeficientes, modos, noise amplification, etc.), y comparaciones entre alternativas. Aprovechen las variantes vistas en clase. Pueden investigar otras variantes de LMS.

Para las simulaciones utilicen, según corresponda, el método de Monte Carlo y/o el set de datos propuesto. Evalúen la complejidad computacional/espacial de las simulaciones, y justifiquen todas las simplificaciones que realicen.

El proyecto tiene tres instancias de presentación:

- En el *design review* presentan el problema, la investigación preliminar, y el enfoque que piensan seguir.
- En la *presentación de proyecto* presentan el desarrollo del problema y las pertinentes conclusiones.
- El *informe final* se presenta una semana después de la presentación. Debe incluir las debidas correcciones comentadas en la presentación. Debe estar escrito en formato IEEE, con la estructura clásica de un trabajo de investigación: introducción (contexto y antecedentes), desarrollo (actividades realizadas), resultados, conclusiones (análisis de resultados, trabajo a futuro) y bibliografía.

No repitan temas entre grupos (salvo que sigan enfoques diferentes).

## Temas sugeridos

- *Denoising de señales EEG*. Deben eliminar el ruido EOG y EMG del EEG. Para las simulaciones pueden usar *EEGdenoisenet*: <https://github.com/ncclabsustech/EEGdenoiseNet>. Investiguen otras bases de datos.
- *Detección de anomalías cardíacas*. Para las simulaciones pueden usar la *MIT-BIH Arrhythmia Database*: <https://physionet.org/content/mitdb/1.0.0>. Para abrir las señales en Python, utilicen *wfdb-python*. Investiguen otras bases de datos.
- *Cancelación de ecos para manos libres*. Para las simulaciones pueden usar la *AEC Challenge Database*: <https://github.com/microsoft/AEC-Challenge>. Investiguen otras bases de datos.
- *Cancelación adaptativa de ruido*. Para las simulaciones pueden usar la *Paths for Active Noise Cancellation Database*: <https://www.iks.rwth-aachen.de/forschung/tools-downloads/databases/paths-for-active-noise-cancellation-development-and-research>. Para abrir las señales en Python, utilicen *scipy.io.loadmat*. Investiguen otras bases de datos.
- *Deconvolución adaptativa de señales sísmicas*. Vean la aplicación 2 del cap. 6 de Haykin. Representen el espectrograma del filtro del error de predicción lineal en función del tiempo. Pueden descargar señales de eventos sísmicos de <https://earthquake.usgs.gov/earthquakes/search>. Para abrir las señales en Python, utilicen *pysac*.

¡Sean creativos! Si algo, que no está en esta lista, les resulta interesante, sacien su curiosidad.

Ante dudas, consulten en Slack.