

UNIVERSIDAD DE LA REPÚBLICA

TESIS DE MAESTRÍA

---

# Codificación de señales multicanal con muestreo irregular

---

*Autor:*

*Pablo Cerveñansky*

*Supervisores:*

*Álvaro Martín*

*Gadiel Seroussi*

Núcleo de Teoría de la Información

Facultad de Ingeniería

29 de junio de 2019



## Capítulo 1

# Título del capítulo

Dataset	#Notación	#Archivos	#Tipos
IRKIS	IRKIS	7	1
NOAA-SST	SST	3	1
NOAA-ADCP	ADCP	3	1
ElNiño	ElNiño	1	7
SolarAnywhere	Solar	4	3
NOAA-SPC-hail	Hail	1	3
NOAA-SPC-tornado	Tornado	1	2
NOAA-SPC-wind	Wind	1	3

TABLA 1.1: Resumen del conjunto de datasets.

En los experimentos realizados se codificaron los datasets combinando estos cuatro parámetros:

- 21 tipos de dato: ver tabla (1.1)
- 13 codificadores:
  - CoderBase
  - CoderPCA-NM y CoderPCA-M
  - CoderAPCA-NM y CoderAPCA-M
  - CoderCA-NM y CoderCA-M
  - CoderPWLH-NM y CoderPWLH-M
  - CoderPWLHInt-NM y CoderPWLHInt-M
  - CoderGampsLimit-NM y CoderGampsLimit-M
- 8 umbrales de error: 0 (sin pérdida), y 1, 3, 5, 10, 15, 20 y 30 (con pérdida)
- 7 tamaños de ventana: 4, 8, 16, 32, 64, 128 y 256.

Algunas consideraciones a tener en cuenta:

- El codificador CoderBase solamente codifica sin pérdida e ignora el parámetro del tamaño de ventana.
- Para los codificadores CoderPCA-NM y CoderPCA-M el tamaño de ventana es fijo, mientras que en el resto de los algoritmos (salvo CoderBase) el tamaño de ventana es variable y el parámetro indica su tamaño máximo.

Para comparar el rendimiento relativo de los algoritmos con ( $M$ ) y sin ( $NM$ ) máscara, utilizamos la siguiente ecuación

$$DiferenciaRelativa(M_S, NM_S) = \begin{cases} 100 \times \frac{NM_S - M_S}{NM_S}, & \text{si } NM_S \neq M_S, \\ 0, & \text{si } NM_S = M_S, \end{cases} \quad (1.1)$$

donde  $M_S$  y  $NM_S$  son los tamaños de los archivos codificados con los respectivos algoritmos. El algoritmo  $M$  logra una mejor tasa de compresión que el algoritmo  $NM$  cuando el resultado de la ecuación 1.1 es mayor a cero. Mientras mayor sea dicho valor, mejor será el rendimiento relativo del algoritmo  $M$  respecto al algoritmo  $NM$ .

[Se considera el dataset de manera global, tomando la ventana óptima global por algoritmo en cada caso. NOTA: para un umbral de error y modo fijo, la ventana óptima global no tiene por qué ser la misma para todos los algoritmos.]

En la tabla 1.2 se muestra un resumen de los resultados obtenidos al comparar el rendimiento relativo de los algoritmos  $NM$  y  $M$  para cada dataset. En la tercera y cuarta columnas aparece el porcentaje de las combinaciones <tipo de dato, codificador, umbral> con las que se obtiene la mejor tasa con cada algoritmo. En la última columna se muestra el rango en el que varía el resultado de la ecuación *DiferenciaRelativa* para dichas combinaciones.

En los datasets que tienen muchos gaps siempre se obtienen mejores tasas al utilizar los algoritmos  $M$ . En cambio, en los datasets sin gaps siempre se tiene mejor rendimiento con los algoritmos  $NM$ . En el dataset con pocos gaps, en cada mitad de las combinaciones se obtienen mejores tasas con algoritmos diferentes.

Dataset	Información	Mejor $NM$	Mejor $M$	Rango <i>DiferenciaRelativa</i>
IRKIS	Muchos gaps	-	100%	(0; 36, 88)
NOAA-SST	Muchos gaps	-	100%	(0; 50, 60)
NOAA-ADCP	Muchos gaps	-	100%	(0; 17, 35)
ElNiño	Muchos gaps	-	100%	(0; 50, 52)
SolarAnywhere	Pocos gaps	49%	51%	(-0, 25; 1, 77)
NOAA-SPC-hail	Sin gaps	100%	-	(-0, 04; 0)
NOAA-SPC-tornado	Sin gaps	100%	-	(-0, 29; 0)
NOAA-SPC-wind	Sin gaps	100%	-	(-0, 12; 0)

TABLA 1.2: Rendimiento relativo de los algoritmos  $NM$  y  $M$ .

Observamos que, en los casos en los que se obtienen mejores tasas con el algoritmo  $NM$ , la diferencia relativa siempre está cerca de 0. En la figura 1.1 vemos que la mejor diferencia relativa a favor de  $NM$  se obtiene para el tipo de dato “Longitude” del dataset NOAA-SPC-tornado, con el codificador CoderAPCA- $NM$  y umbral de error 30%. Como se observa en la tabla 1.2, dicho valor es -0, 29.

Por otro lado, cuando se logran mejores tasas con el algoritmo  $M$  las diferencias relativas son mucho mayores, alcanzando un máximo de 50,60 para el tipo de dato “VWC” del dataset NOAA-SST. En la figura 1.2 vemos que dicho resultado se obtiene con el codificador CoderPCA- $M$  y umbral de error 30%.

Teniendo en cuenta los resultados presentados, si quisiéramos codificar un dataset que a priori supiéramos tiene muchos gaps, obviamente nos convendría utilizar el algoritmo  $M$ . Pero aún si el dataset no tuviera gaps, la diferencia de rendimiento a favor del algoritmo  $NM$  sería despreciable. Como el algoritmo  $M$  es más robusto y funciona mejor en general, en las próximas secciones nos vamos a enfocar en su estudio.

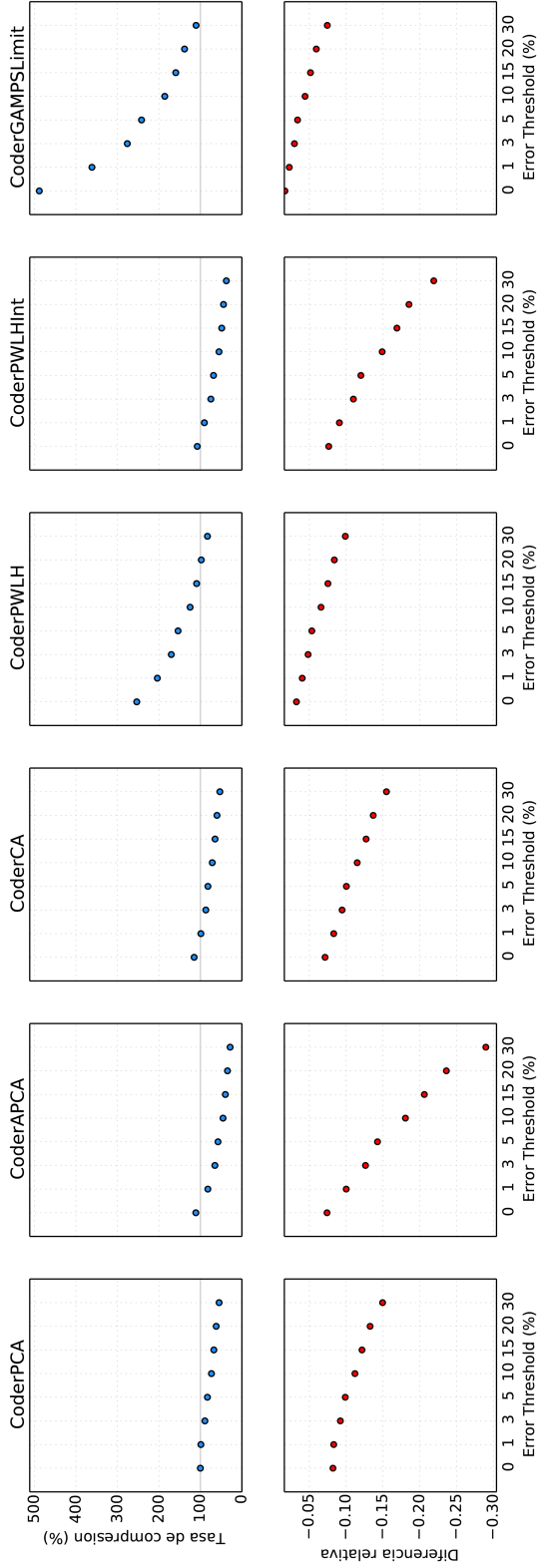


FIGURA 1.1: Tasa de compresión y Diferencia relativa para las distintas combinaciones <codificador, umbral> para el tipo de dato “Longitude” del dataset NOAA-SPC-tornado.

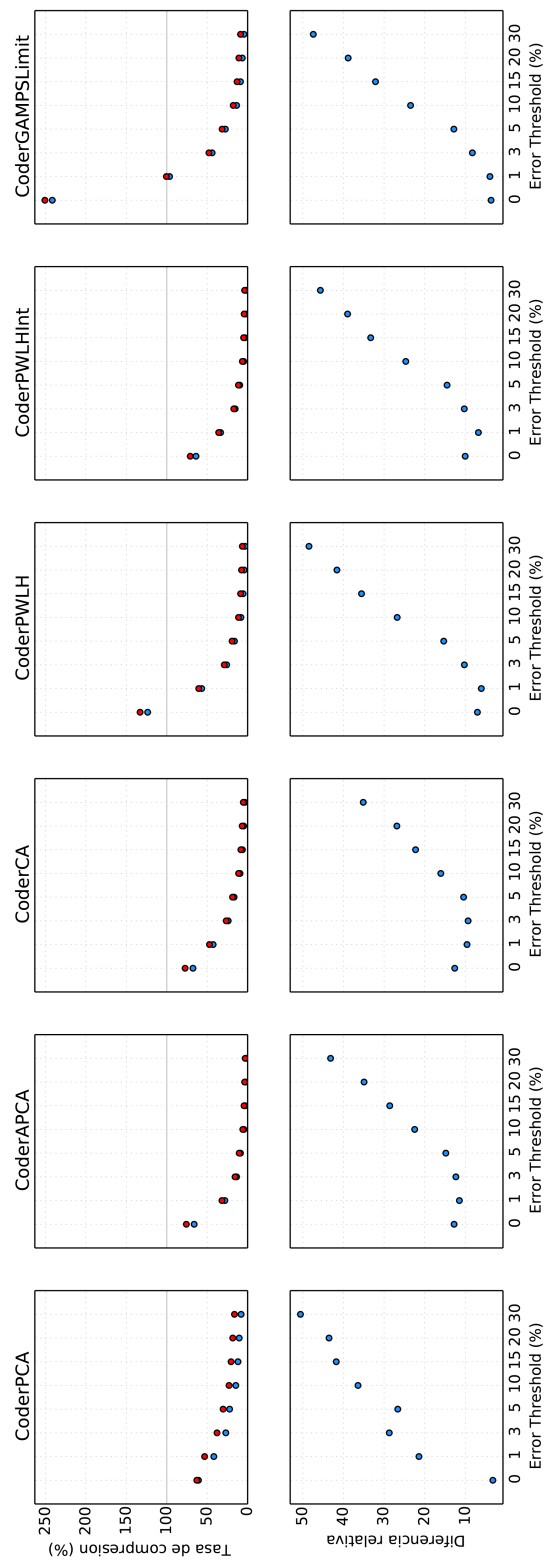


FIGURA 1.2: Tasa de compresión y Diferencia relativa para las distintas combinaciones <codificador, umbral> para el tipo de dato “VWC” del dataset NOAA-SST.

## HECHO INFORME:

- Elegir nomenclatura para los dos distintos modos de ejecución => CoderPCA-NM (sin máscara) y CoderPCA-M (con máscara).
- Realizar un análisis cuantitativo para saber qué tanto mejor comprime el modo MM=0 en los pocos casos en los que funciona mejor que el modo MM=3. Vimos que esos casos se dan en los datasets con pocos o ningún gap, y la diferencia en las tasas de compresión es mínima. En cambio, cuando hay gaps en los datasets, la diferencia relativa de rendimiento a favor del modo MM=3 es mayor. Escribir un párrafo con dicho análisis, incluyendo alguna gráfica como ejemplo.
- Agregar tabla con resumen de los datasets - ver AVANCES / DUDAS (13)

## TODO INFORME:

- Poner las gráficas horizontales, 3 arriba y 3 abajo.
- Vimos que en los datasets sin gaps, en general para todas las combinaciones <tipo de dato, algoritmo> la diferencia relativa no crece al aumentar el umbral de error. Escribir un párrafo explicando el por qué de este comportamiento.
- Mencionar experimentos ventana local vs ventana global. (ver minuta de la reunión del lunes 10/06/2019).
- Mencionar relación de compromiso entre el umbral y la tasa de compresión: al aumentar el umbral mejor la tasa de compresión (lógico).
- Mencionar que CoderSlideFilter no tiene en cuenta el parámetro con el tamaño máximo de la ventana.
- Agregar tabla con resumen de los algoritmos.
- Subir todo el material complementario en un link (después referirlo en el informe)

## TODO CÓDIGO:

- Para los experimentos sin máscara no se están considerando los datos para los algoritmos CoderFractalRestampling y CoderSlideFilter.
- Agregar tests para MM=3.
- Al ejecutar los algoritmos GAMPS/GAMPSLimit sobre el dataset de "El Niño" (546 columnas) tengo problemas de memoria en Ubuntu, pero no en la Mac.
- Universalizar algoritmo
- Modificar GAMPS/GAMPSLimit para que utilice floats (4 bytes) en vez de doubles (8 bytes). De todas maneras, no creo que esto cambie los resultados de manera significativa, ya que aun si la cantidad de bits utilizados al codificar con GAMPS/GAMPSLimit fuera la mitad, en ningún caso superaría la tasa obtenida con el mejor codificador.