Universidad de la República

Tesis de Maestría

Coding of Multichannel Signals with Irregular Sampling

Autor:
Pablo Cerveñansky

Supervisores:
Álvaro Martín
Gadiel Seroussi

Núcleo de Teoría de la Información
Facultad de Ingeniería

21 de mayo de 2020

Chapter 1

Datasets

1.1 Introduction

Explain why / how every dataset was transformed into a common format.
Show example csv (describe header, data rows, etc.)

Dataset	#Files	#Types	Data Types
IRKIS	7	1	VWC
SST	3	1	SST
ADCP	3	1	Vel
Solar	4	3	GHI, DNI, DHI
ElNino	1	7	Lat, Long, Zonal Winds, Merid. Winds, Humidity, Air Temp., SST
Hail	1	3	Lat, Long, Size
Tornado	1	2	Lat, Long
Wind	1	3	Lat, Long, Speed

Tabla 1.1: Datasets overview.

1.2 IRKIS

1.3 SST

1.4 ADCP

1.5 ElNino

1.6 Solar

1.7 Hail

1.8 Tornado

1.9 Wind

Chapter 2

Algorithms

2.1 Introduction

- Add two papers as biography
- Explain the Arithmetic Coder (add website as bibliography)
- Explain KT estimator
- Add references to C implementation of the coders
- Add references to each coder
- Explain MASK_MODE macro in the C++ code.

Some of the implementation details are not shown in the pseudocode.

2.2 Base

```
input  : in: csv data file to be coded
output: out: binary file coded with algorithm Base
out = new_binary_file()
out.code_integer(INT_BASE, 8)
out.code_header(in.header)
out.code_integer(in.count_data_rows(), 24)
foreach column in in.columns do
    foreach entry in column.entries do
        if value == NO_DATA then
            value = column.no_data_int
        else
            value = entry + column.offset
        end
        out.code_integer(value, column.total_bits)
    end
end
out.close_file()
```

FIGURE 2.1: Algorithm Base coder pseudocode.

```
input  : in: binary file coded with algorithm Base
output: out: decoded csv data file
out = new_csv_file()
int_algo = in.decode_integer(8)
out.decode_header(in)
count_data_rows = in.decode_integer(24)
if int_algo == INT_BASE then
    foreach column in out.columns do
        foreach entry in column.entries do
            value = in.decode_integer(column.total_bits)
            if value == column.no_data_int then
                out.write_string(NO_DATA)
            else
                out.write_string(value − column.offset)
            end
        end
    end
else
    . . . // if in was coded with a different algorithm
end
out.close_file()
```

FIGURE 2.2: Algorithm decoder coder pseudocode.

2.3 PCA

```
input  : in: csv data file to be coded, w: window size parameter, e: threshold parameter
output: out: binary file coded with algorithm PCA
out = new_binary_file()
out.code_integer(INT_PCA, 8)
out.code_integer($w - 1, 8$)
out.code_header(in.header)
out.code_integer(in.count_data_rows(), 24)
foreach column in in.columns do
    err = column.error_for_threshold_parameter(e)
    win = new_window()
    foreach entry in column.entries do
        win.push(entry)
        if win.size $<$ w then
            continue
        end
        if win.all_entries_are_no_data then
            out.code_bit(0)
            out.code_integer(column.no_data_int, column.total_bits)
        else if win.all_entries_are_integers and $|win.\text{max} - win.\text{min}| \leq 2 * err$ then
            value = win.min + win.max
            if $value \neq 0$ then
                value = value/2
            end
            out.code_bit(0)
            out.code_integer(value + column.offset, column.total_bits)
        else
            out.code_bit(1)
            foreach win_val in win.values do
                if win_val == NO_DATA then
                    value = column.no_data_int
                else
                    value = win_val + column.offset
                end
                out.code_integer(value, column.total_bits)
            end
        end
        win = new_window()
    end
end
out.close_file()
```

FIGURE 2.3: Algorithm PCA coder pseudocode.

2.4 APCA

```
input  : in: csv data file to be coded, w: window size parameter, e: threshold parameter
output: out: binary file coded with algorithm APCA
out = new_binary_file()
out.code_integer(INT_PCA, 8)
out.code_integer($w - 1, 8$)
out.code_header(in.header)
out.code_integer(in.count_data_rows(), 24)
foreach column in in.columns do
    err = column.error_for_threshold_parameter(e)
    win = new_window()
    foreach entry in column.entries do
        if win.size == 0 then
            win.push(entry)
            continue
        end
        code_window = false
        if win.size == w then
            code_window = true
        else if win.all_entries_are_no_data then
            if entry == NO_DATA then
                win.push(entry)
            else
                code_window = true
            end
        else
            // win.all_entries_are_integers
            if entry == NO_DATA then
                code_window = true
            else
                win.push(entry)
                if $|win.max - win.min| \leq 2 * err$ then
                    value = win.min + win.max
                    if $value \neq 0$ then
                        value = value/2
                    end
                    win.code_value = value
                else
                    entry = win.unpush()
                    code_window = true
                end
            end
        end
        if code_window then
            out.code_integer(win.size $- 1, \log_2 w$)
            if win.all_entries_are_no_data then
                value = column.no_data_int
            else
                value = win.code_value + column.offset
            end
            out.code_integer(value, column.total_bits)
            win = new_window()
            win.push(entry)
        end
    end
end
out.close_file()
```

Figure 2.4: Algorithm APCA coder pseudocode.

2.5 CA

2.6 PWLH and PWLHInt

2.7 GAMPS and GAMPSLimit

Ver los siguientes documentos:

- [08] AVANCES / DUDAS
- [09] AVANCES / DUDAS
- [10] AVANCES / DUDAS
- [11] AVANCES / DUDAS
- [12] AVANCES / DUDAS

2.8 FR

2.9 SF

Un punteo de los cambios más importantes respecto a la versión anterior del informe. Los cambios de la entrega del 12.03.2020 están en azul.

Cambios generales:

- Sustituyo todos los “CoderABC” por “ABC” (en el texto y en las gráficas)

Cambios en las gráficas:

- Agrego (%) después de CR y RD
- Cambié los markers en las gráficas de CR (Section 3.2)
- Cambié los markers y los colores en las gráficas de Window parameter (Section 3.3)

Introduction:

- Agrego una frase de conclusión para las sections 3.2 y 3.3

Section 3.2:

- Sustituyo todos los “mode” por “variant” / En las definiciones cambio “||” por “|”
- Mejoré explicación de la Table 3.1.
- Reescribí la descripción abajo de las gráficas
- El párrafo anterior a Table 3.1 lo moví a la misma página de la tabla. Por un lado me parece mejor que este párrafo esté en la misma página de la tabla (sino quedan las dos páginas de gráficas en el medio). Por otro lado no queda muy bien tanto espacio en blanco en la página anterior a las gráficas.

Section 3.3:

- Las referencias $a \in A$ las debería cambiar. En vez de A (definido al comienzo de la Section 3.2) debería definir otro conjunto que incluye solamente las variantes con máscaras de A y el algoritmo FR (que solo tiene variante con máscara y que no había tenido en cuenta en las comparaciones de la Section 3.2).

Section 3.4:

- Empiezo a escribir la sección. Agrego las tablas.

Chapter 3

Experimental Results

In this chapter we present our experimental results. The main goal of our experiments is to analyze the performance of each of the coding algorithms presented in Chapter 2, by encoding the various datasets introduced in Chapter 1. In Section 3.1 we describe our experimental setting, defining the evaluated combinations of algorithms and parameter values, and the figures of merit used for comparison. In Section 3.2 we compare the compression performance of the masking and non-masking variants for each coding algorithm. The results suggest that the masking variant is more robust and performs better in general. In Section 3.3 we analyze the extent to which the window size parameter impacts on the performance of the algorithms. The results indicate that the impact of using the global window instead of the local window on the compression performance of the coding algorithms is rather small. In Section 3.4 we compare the performance of the different algorithms among each other and with the general purpose compression algorithm gzip. ...

3.1 Experimental Setting

We evaluate the compression performance of all the coding algorithms presented in Chapter 2 on the datasets described in Chapter 1. For each algorithm we test both the masking and the non-masking variants (except for *Base*, *FR*, and *SF*, which do not admit a masking variant).

We also test several combinations of algorithm parameters. Specifically, for the algorithms that admit a window size parameter w (every algorithm except *Base* and *SF*), we test all the values of w in the set $W = \{4, 8, 16, 32, 64, 128, 256\}$. For the encoders that admit a lossy compression mode with a threshold parameter e (every encoder except *Base*), we test all the values of e in the set $E = \{1, 3, 5, 10, 15, 20, 30\}$, where each threshold is expressed as a percentage fraction of the standard deviation of the data being encoded. For example, for certain data with a standard deviation of 20, taking $e = 10$ implies that the lossy compression allows for a maximal per-sample distortion of 2 sampling units.

Definition 3.1.1. We refer to a specific combination of a coding algorithm variant and its parameter values as a *coding algorithm instance (CAI)*. We define CI as the set of all the CAIs obtained by combining each of the algorithm variants presented in Chapter 2 with the parameter values (from W and E) that are suitable for that algorithm variant. We denote by $c_{<a,w,e>}$ the CAI obtained by setting a window size parameter equal to w and a threshold parameter equal to e on algorithm variant a.

We assess the compression performance of a CAI mainly through the compression ratio, which we define next. For this definition, we regard *Base* as a trivial CAI that serves as a base ground for compression performance comparison (recall the definition of algorithm *Base* from Section 2.2).

Definition 3.1.2. Let f be a file and z a data type of a certain dataset. We define f_z as the subset of data of type z from file f. For example, for the dataset Hail, the data type z may be Latitude, Longitude, or Size.

Definition 3.1.3. Let f be a file and z a data type of a certain dataset. Let $c \in CI$ be a CAI. We define $|c(z, f)|$ as the size of the resulting file obtained when using coding f_z with c.

Definition 3.1.4. The *compression ratio (CR)* of a CAI $c \in CI$ for the data type z of a certain file f is the fraction of $|c(z, f)|$ with respect to $|Base(z, f)|$, i.e.,

$$CR(c, z, f) = \frac{|c(z, f)|}{|Base(z, f)|}. \tag{3.1}$$

Notice that smaller values of CR imply better performance. Thus, our main goals are to analyze which CAIs yield the smallest values in (3.1) for the different data types, and to study how the CR depends on the different algorithm variants and parameter values.

To compare the compression performance between a pair of CAIs we calculate the relative difference, which we define next.

Definition 3.1.5. The *relative difference (RD)* between a pair of CAIs $c_1, c_2 \in CI$ for the data type z of a certain file f is given by

$$RD(c_1, c_2, z, f) = 100 \times \frac{|c_2(z, f)| - |c_1(z, f)|}{|c_2(z, f)|}. \tag{3.2}$$

Notice that c_1 has a better performance than c_2 if (3.2) is positive.

In some of our experiments we consider the performance of algorithms on complete datasets, rather than individual files. With this in mind, we extend the definitions 3.1.3–3.1.5 to datasets as follows.

Definition 3.1.6. Let z be a data type of a certain dataset d. We define $F(d, z)$ as the set of files f from dataset d for which f_z is not empty.

Definition 3.1.7. Let z be a data type of a certain dataset d. Let $c \in CI$ be a CAI. We define $|c(z, d)|$ as

$$|c(z, d)| = \sum_{f \in F(d,z)} |c(z, f)|. \tag{3.3}$$

Definition 3.1.8. The *compression ratio (CR)* of a CAI $c \in CI$ for the data type z of a certain dataset d is given by

$$CR(c, z, d) = \frac{|c(z, d)|}{|Base(z, d)|}. \tag{3.4}$$

Definition 3.1.9. The *relative difference (RD)* between a pair of CAIs $c_1, c_2 \in CI$ for the data type z of a certain dataset d is given by

$$RD(c_1, c_2, z, d) = 100 \times \frac{|c_2(z, d)| - |c_1(z, d)|}{|c_2(z, d)|}. \tag{3.5}$$

3.2 Comparison of Masking and Non-Masking Variants

In this section, we compare the compression performance of the masking and non-masking variants of each of the evaluated algorithms that admit both; we denote by A this set of algorithm variants. Specifically, we compare:

- *PCA-M* against *PCA-NM*
- *APCA-M* against *APCA-NM*
- *CA-M* against *CA-NM*
- *PWLH-M* against *PWLH-NM*
- *PWLHInt-M* against *PWLHInt-NM*
- *GAMPSLimit-M* against *GAMPSLimit-NM*.

Notation 3.2.1. We denote by a_M and a_{NM} the masking and non-masking variants of an algorithm $a \in A$.

For each algorithm and each threshold parameter, we compare the performance of the masking and non-masking variants of the algorithm. For the purpose of this comparison, we choose the most favorable window size for each variant, in the sense of the following definition.

Definition 3.2.1. The *optimal window size (OWS)* of a coding algorithm $a \in A$ and a threshold parameter $e \in E$, for the data type z of a certain dataset d is given by

$$OWS(a, e, z, d) = \underset{w \,\in\, W}{\arg\min} \left\{ CR(c_{<a,w,e>}, z, d) \right\}, \tag{3.6}$$

where we break ties in favor of the smallest window size.

For each data type z of each dataset d, and each coding algorithm $a \in A$ and threshold parameter $e \in E$, we calculate the RD between $c_{<a_M,w_M^*,e>}$ and $c_{<a_{NM},w_{NM}^*,e>}$, as defined in (3.5), where $w_M^* = \text{OWS}(a_M, e, z, d)$ and $w_{NM}^* = \text{OWS}(a_{NM}, e, z, d)$.

As an example, in figures 3.1 and 3.2 we show the CR and the RD, as a function of the error threshold, obtained for two data types of two different datasets. Figure 3.1 shows the results for the data type “SST” of the dataset SST, and Figure 3.2 shows the results for the data type “Longitude” of the dataset Tornado. In Figure 3.1 we observe a large RD favoring the masking variant for all tested algorithms. On the other hand, in Figure 3.2 we observe that the non-masking variant outperforms the masking variant for all algorithms. We notice, however, that the RD is very small in the latter case.

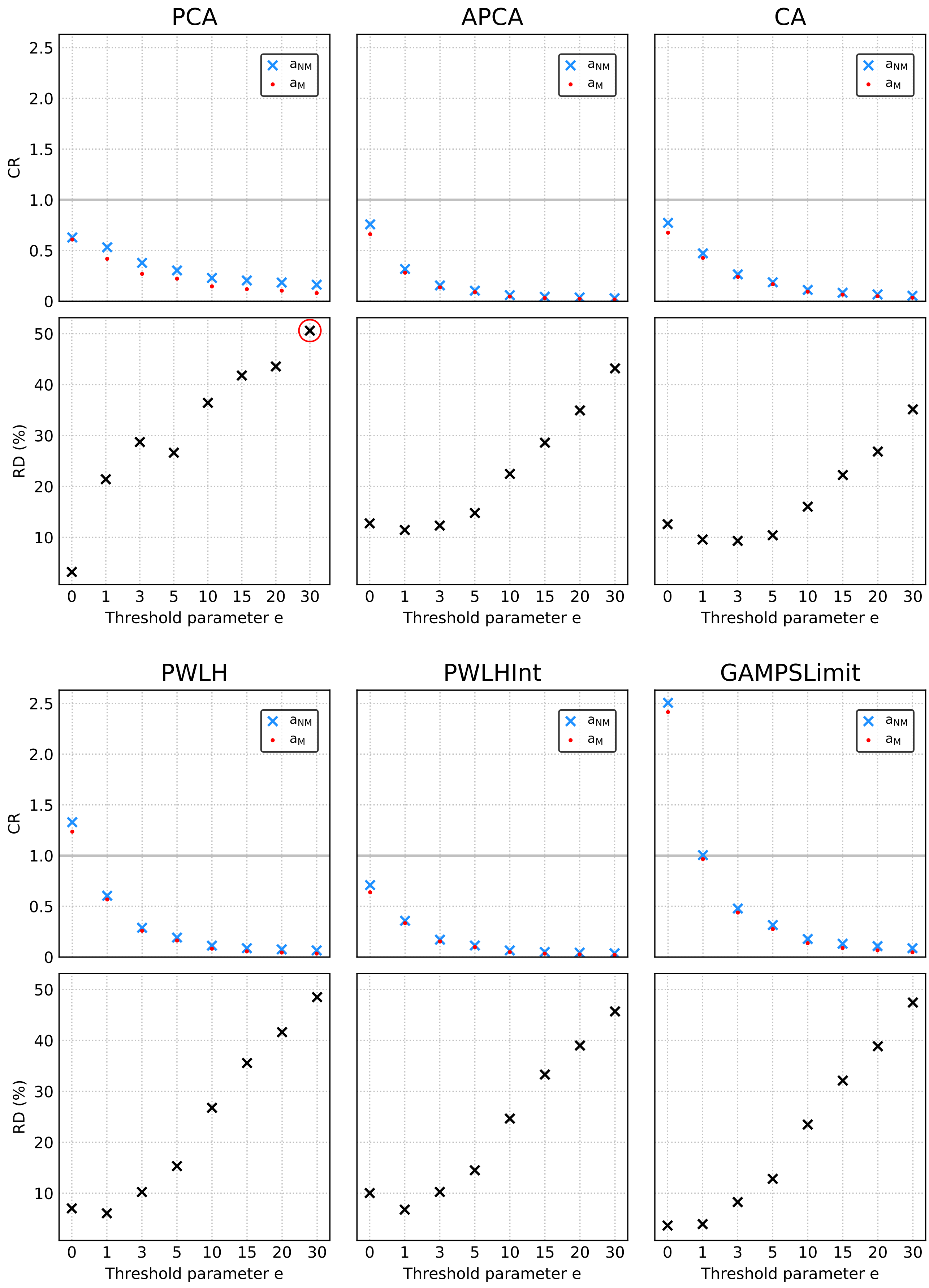

FIGURE 3.1: CR and RD plots for every pair of algorithm variants $a_M, a_{NM} \in A$, for the data type "SST" of the dataset SST. In the RD plot for algorithm PCA we highlight with a red circle the marker for to the maximum value (50.60%) obtained for all the tested CAIs.

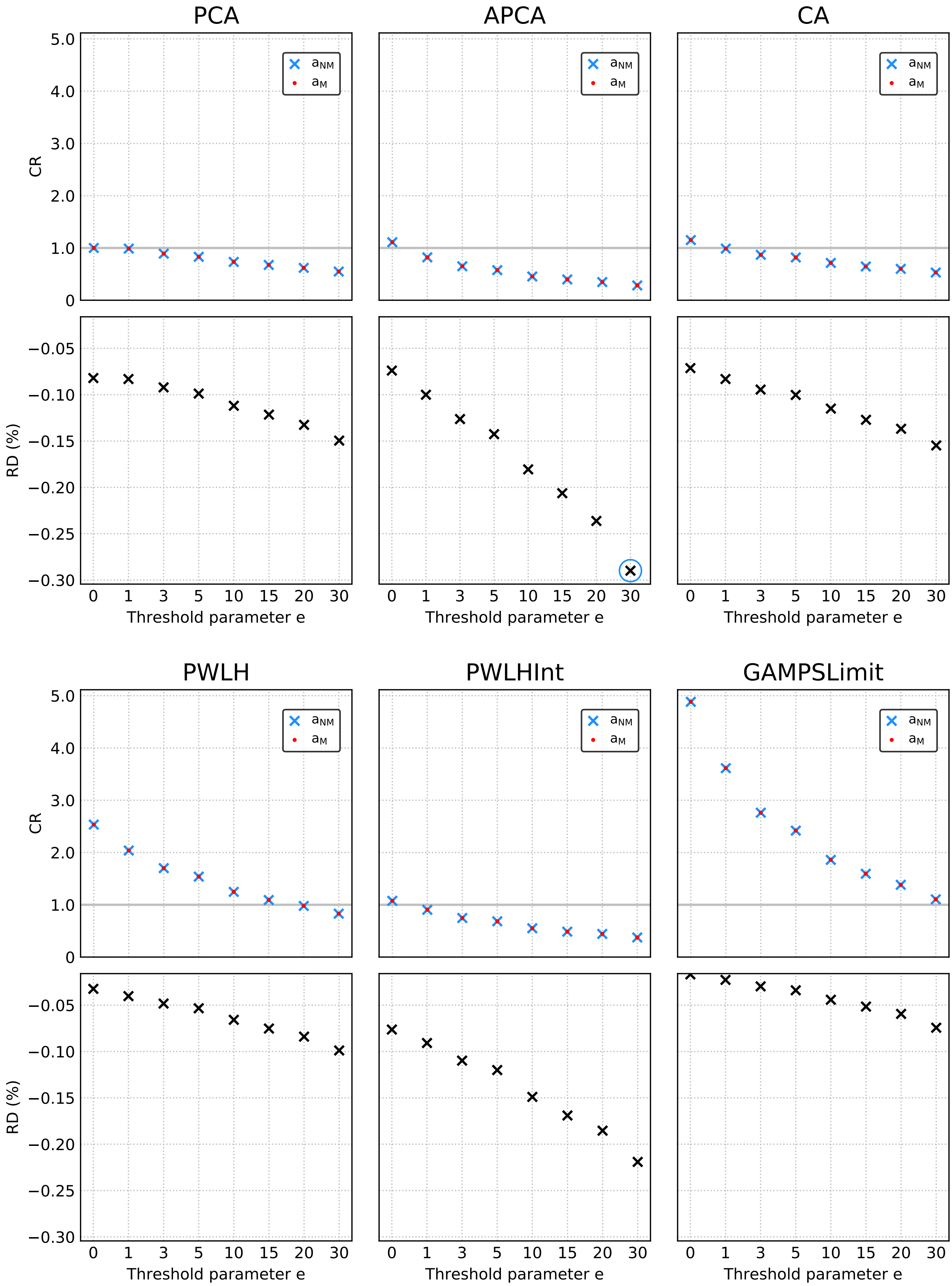

FIGURE 3.2: CR and RD plots for every pair of algorithm variants $a_M, a_{NM} \in A$, for the data type "Longitude" of the dataset Tornado. In the RD plot for algorithm APCA we highlight with a blue circle the marker for to the minimum value (-0.29%) obtained for all the tested CAIs.

We analyze the experimental results to compare the performance of the masking and non-masking variants of each algorithm. For each data type, we iterate through each algorithm $a \in A$, and each threshold parameter $e \in E$, and we calculate the RD between the CAIs $c_{<a_M, w^*_M, e>}$ and $c_{<a_{NM}, w^*_{NM}, e>}$, obtained by setting the OWS for the masking variant a_M and the non-masking variant a_{NM}, respectively. Since we consider 8 threshold parameters and there are 6 algorithms, for each data type we compare a total of 48 pairs of CAIs. Table 3.1 summarizes the results of these comparisons, aggregated by dataset. The number of pairs of CAIs evaluated for each dataset depends on the number of different data types it contains.

Dataset	**Dataset Characterstic**	**Cases where a_M outperforms a_{NM} (%)**	**RD (%) Range**
IRKIS	Many gaps	48/48 (100%)	(0; 36.88]
SST	Many gaps	48/48 (100%)	(0; 50.60]
ADCP	Many gaps	48/48 (100%)	(0; 17.35]
ElNino	Many gaps	336/336 (100%)	(0; 50.52]
Solar	Few gaps	73/144 (50.7%)	[-0.25; 1.77]
Hail	No gaps	0/144 (0%)	[-0.04; 0)
Tornado	No gaps	0/96 (0%)	[-0.29; 0)
Wind	No gaps	0/144 (0%)	[-0.12; 0)

Tabla 3.1: RD between the masking and non-masking variants of each algorithm. The results are aggregated by dataset. In the last column we highlight the maximum (red) and minimum (blue) values taken by the RD.

Consider, for example, the results for the dataset Tornado, in the 7th row. The second column shows that there are no gaps in any of the data types of the dataset (recall the dataset description from Table 1.1). Since the dataset has two data types, we compare a total of $2 \times 48 = 144$ pairs of CAIs. The third column reveals that in none of these comparisons the masking variant a_M outperforms the non-masking variant a_{NM}, i.e. the RD is always negative. The last column shows the range for the values attained by the RD for those tested CAIs.

Observing the last column of Table 3.1, we notice that in every case in which the non-masking variant performs best, the RD is close to zero. The minimum value it takes is -0.29%, which is obtained for the data type “Longitude” of the dataset Tornado, with algorithm APCA, and error parameter $e = 30$. In Figure 3.2 we highlight the marker associated to this minimum with a blue circle. On the other hand, we also notice that for the datasets in which the masking variant performs best, the RD reaches high absolute values. The maximum (50.60%) is obtained for the data type “VWC” of the dataset SST, with algorithm PCA, and error parameter $e = 30$, which is highlighted in Figure 3.1 with a red circle.

The experimental results presented in this section suggest that if we were interested in compressing a dataset with many gaps, we would benefit from using the masking variant of an algorithm, a_M. However, even if the dataset didn't have any gaps, the performance would not be significantly worse than that obtained by using the non-masking variant of the algorithm, a_{NM}. Therefore, since masking variants are, in general, more robust in this sense, in the sequel we focus on these variants.

3.3 Window Size Parameter

In this section, we analyze the extent to which the window size parameter impacts on the performance of the coding algorithms. We only consider the four datasets that consist of multiple files, i.e. IRKIS, SST, ADCP and Solar. For each file, we compare the compression performance when using the OWS for the dataset, as defined in (3.6), and the LOWS for the file, defined next.

Definition 3.3.1. The *local optimal window size (LOWS)* of a coding algorithm $a \in A$ and a threshold parameter $e \in E$, for the data type z of a certain file f is given by

$$LOWS(a, e, z, f) = \underset{w \,\in\, W}{\arg\min} \left\{ CR(c_{<a,w,e>}, z, f) \right\}, \tag{3.7}$$

where we break ties in favor of the smallest window size.

For each data type z of each dataset d, and each file $f \in F(d, z)$, coding algorithm $a \in A$ and threshold parameter $e \in E$, we calculate the RD between $c_{<a,w^*_{global},e>}$ and $c_{<a,w^*_{local},e>}$, as defined in (3.2), where $w^*_{global} = \text{OWS}(a, e, z, d)$ and $w^*_{local} = \text{LOWS}(a, e, z, f)$. In what follows, we refer to w^*_{global} and w^*_{local} as the OWS and the LOWS , respectively.

As an example, in figures 3.3 and 3.4 we show the OWS and the LOWS , and the RD, as a function of the threshold, obtained for the data type “VWC”, for two different files of the dataset IRKIS. Figure 3.3 shows the results for the file “vwc_1202.dat.csv”, and Figure 3.4 shows the results for “vwc_1203.dat.csv”. Observe that the OWS values are the same for both figures, which is expected, since both are obtained from the same data type of the same dataset.

In Figure 3.3 we notice, for instance, that in the algorithm APCA the OWS and LOWS values match for every threshold parameter e, except 3 and 10. The OWS is larger than the LOWS when $e = 3$, but it is smaller when $e = 10$. In these two cases, the RD values are 1.52 and 1.76, respectively. Notice that the RD is non-negative in every plot, which makes sense, since the CR obtained with the OWS cannot be lower than the CR obtained with the LOWS .

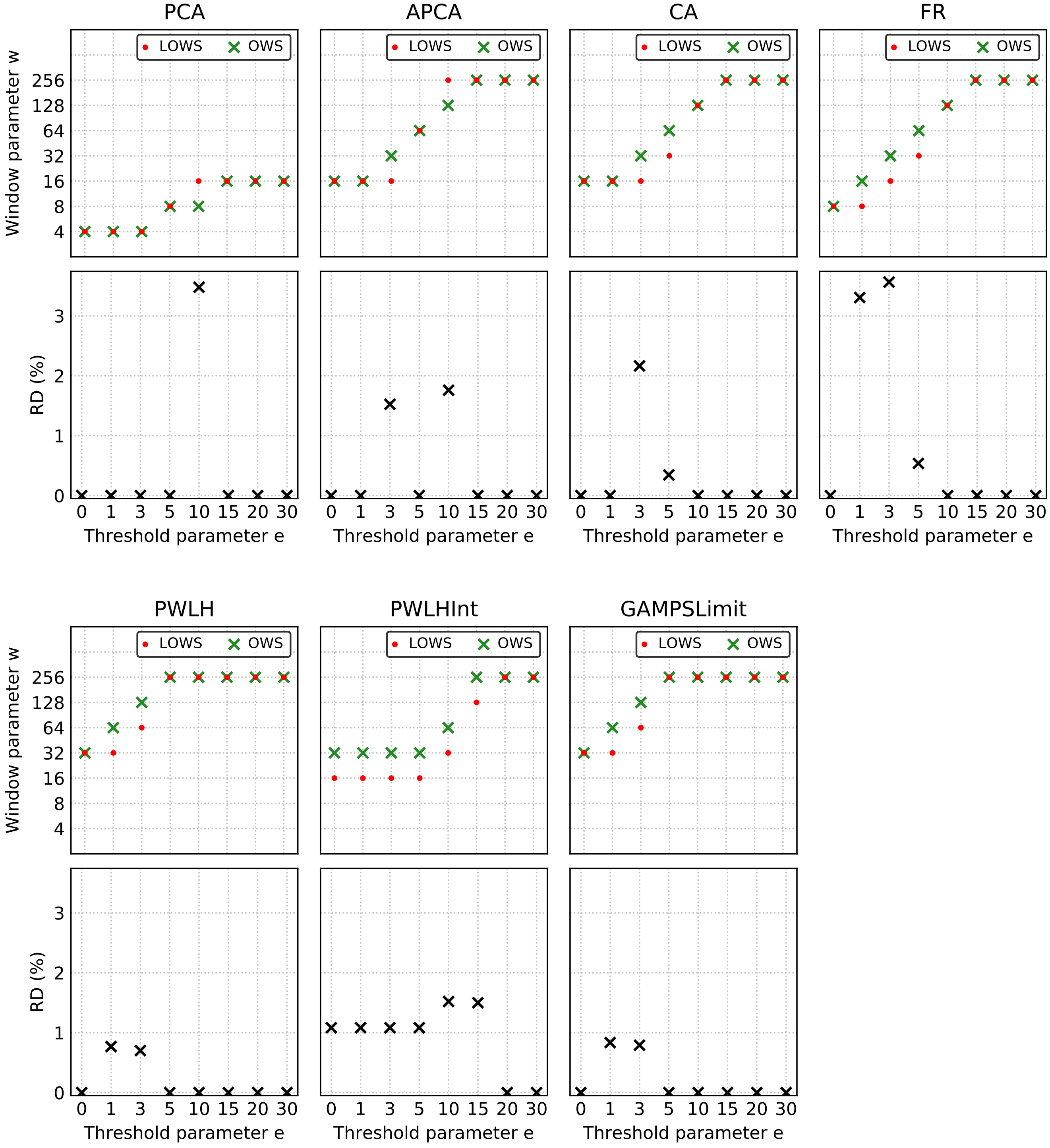

Figure 3.3: OWS and LOWS, and RD plots for every algorithm, for the data type “VWC” of the file “vwc_1202.dat.csv” of the dataset IRKIS.

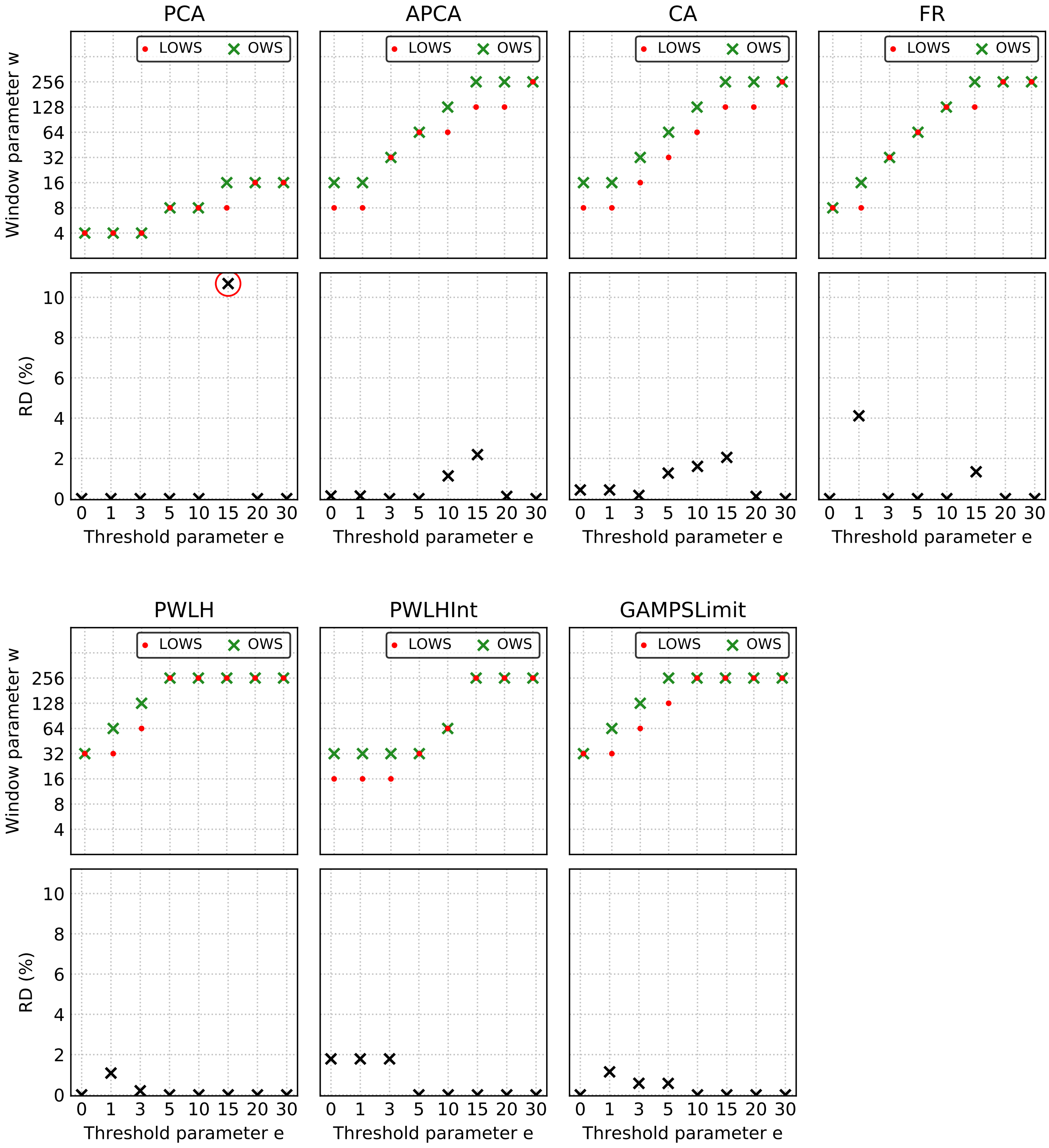

FIGURE 3.4: OWS and LOWS, and RD plots for every algorithm, for the data type "VWC" of the file "vwc_1203.dat.csv" of the dataset IRKIS. In the RD plot for algorithm PCA we highlight with a red circle the marker for to the maximum value (10.68%) obtained for all the tested CAIs.

We analyze the experimental results to evaluate the impact of using the OWS instead of the LOWS on the compression performance of the tested coding algorithms. For each algorithm, we iterate through each threshold parameter, and each data type of each file, and we calculate the RD between the CAI with the OWS and the CAI with the LOWS . Since we consider 8 threshold parameters and there are 13 files with a single data type and 4 files with 3 different data types each, for each algorithm we compare a total of $8 \times (13 + 4 \times 3) = 200$ pairs of CAIs. Table 3.2 summarizes the results of these comparisons, aggregated by algorithm and the range to which the RD belongs.

	RD (%) Range				
Algorithm	**0**	**(0,1]**	**(1,2]**	**(2,5]**	**(5,11]**
PCA	186 (93%)	4 (2%)	3 (1.5%)	2 (1%)	5 (2.5%)
APCA	174 (87%)	13 (6.5%)	7 (3.5%)	6 (3%)	0
CA	172 (86%)	16 (8%)	6 (3%)	6 (3%)	0
FR	171 (85.5%)	14 (7%)	8 (4%)	7 (3.5%)	0
PWLH	184 (92%)	13 (6.5%)	3 (1.5%)	0	0
PWLHInt	173 (86.5%)	9 (4.5%)	13 (6.5%)	4 (2%)	1 (0.5%)
GAMPSLimit	182 (91%)	16 (8%)	2 (1%)	0	0
Total	1,242 (88.7%)	85 (6.1%)	42 (3%)	25 (1.8%)	6 (0.4%)

Tabla 3.2: RD between the OWS and LOWS variants of each CAI.
The results are aggregated by algorithm and the range to which the RD belongs.

For example, consider the results for algorithm CA, in the third row. The first column indicates that the RD is equal to 0 for exactly 172 (86%) of the 200 evaluated pairs of CAIs for algorithm CA. The second column reveals that for 16 pairs of CAIs (8%), the RD takes values greater than 0 and less than or equal to 1%. The remaining three columns cover other ranges of RD. Notice that for every row (except the last one), the values add up to a total of 200, since we compare exactly 200 pairs of CAIs for each algorithm.

The last row of Table 3.2 is obtained by adding the values of the previous rows, which combines the results for all algorithms. We notice that in 88.7% of the total number of evaluated pairs of CAIs, the RD is equal to 0. In these cases, in fact, the OWS and the LOWS coincide. In 97.8% of the cases, the RD is less than or equal to 2%. This means that, for the vast majority of CAI pairs, either the OWS and the LOWS match or they yield roughly the same compression performance. This result suggests that we could fix in advance the window size parameter, for example by optimizing over a training set, without compromising the performance of the coding algorithm. This is relevant, since calculating the LOWS for a file is, in general, computationally expensive.

We notice that there are only 6 cases (0.4%) in which the RD falls in the range (5, 11], most of which (5 cases) involve the algorithm PCA. The maximum value taken by RD (10.68%) is obtained for the data type "VWC" of the file "vwc_1203.dat.csv" of the dataset SST, with algorithm PCA, and error parameter $e = 15$. In Figure 3.4 we highlight this maximum value with a red circle. In this case, the OWS is 16 and the LOWS is 8. According to these results, the performance of algorithm PCA seems to be more sensible to the window size parameter than the rest of the algorithms. Except for these few cases, we observe that, in general, the impact of using the OWS instead of the LOWS on the compression performance of coding algorithms is rather small. Therefore, in the following section, in which we compare the algorithms performance, we always use the OWS.

3.4 Algorithms Performance

In this section we compare the compression performance of the coding algorithms presented in Chapter 2, by encoding the various datasets introduced in Chapter 1. We begin by comparing the encoders among each other and later we compare them with gzip, a popular lossless compression algorithm. We analyze the performance of the algorithms on complete datasets (not individual files), so we always apply definitions 3.1.6–3.1.9. Following the results obtained in sections 3.2 and 3.3, we only consider the masking variants of the evaluated algorithms, and we always use the OWS.

For each data type z of each dataset d, and each coding algorithm $a \in A$ and threshold parameter $e \in E$, we calculate the CR of $c_{<a,w^*_{global},e>}$, as defined in (3.4), where $w^*_{global} = \mathrm{OWS}(a, e, z, d)$. The following definition is useful for analyzing which CAI obtains the best compression result for a specific data type.

Definition 3.4.1. Let z be a data type of a certain dataset d, and let $e \in E$ be a threshold parameter. The *best CAI* for z, d, e is the CAI c^b_e that minimizes the CR among all CAIs from CI.

Our experiments include a total of 21 data types, in 8 datasets. As an example, in Figure 3.5 we show the CR and the OWS, as a function of the threshold, obtained for each algorithm for the data type "SST" of the dataset ElNino. For each threshold parameter $e \in E$, we use blue circles to highlight the markers for the minimum CR value and the best window size parameter (in the plots corresponding to the best algorithm). For instance, for $e = 0$, the minimum CR is 0.33, and the best algorithm is PCA with a window size equal to 256. For the remaining seven threshold parameters, the blue circles indicate that the best algorithm is always APCA with a window size ranging from 4 to 32.

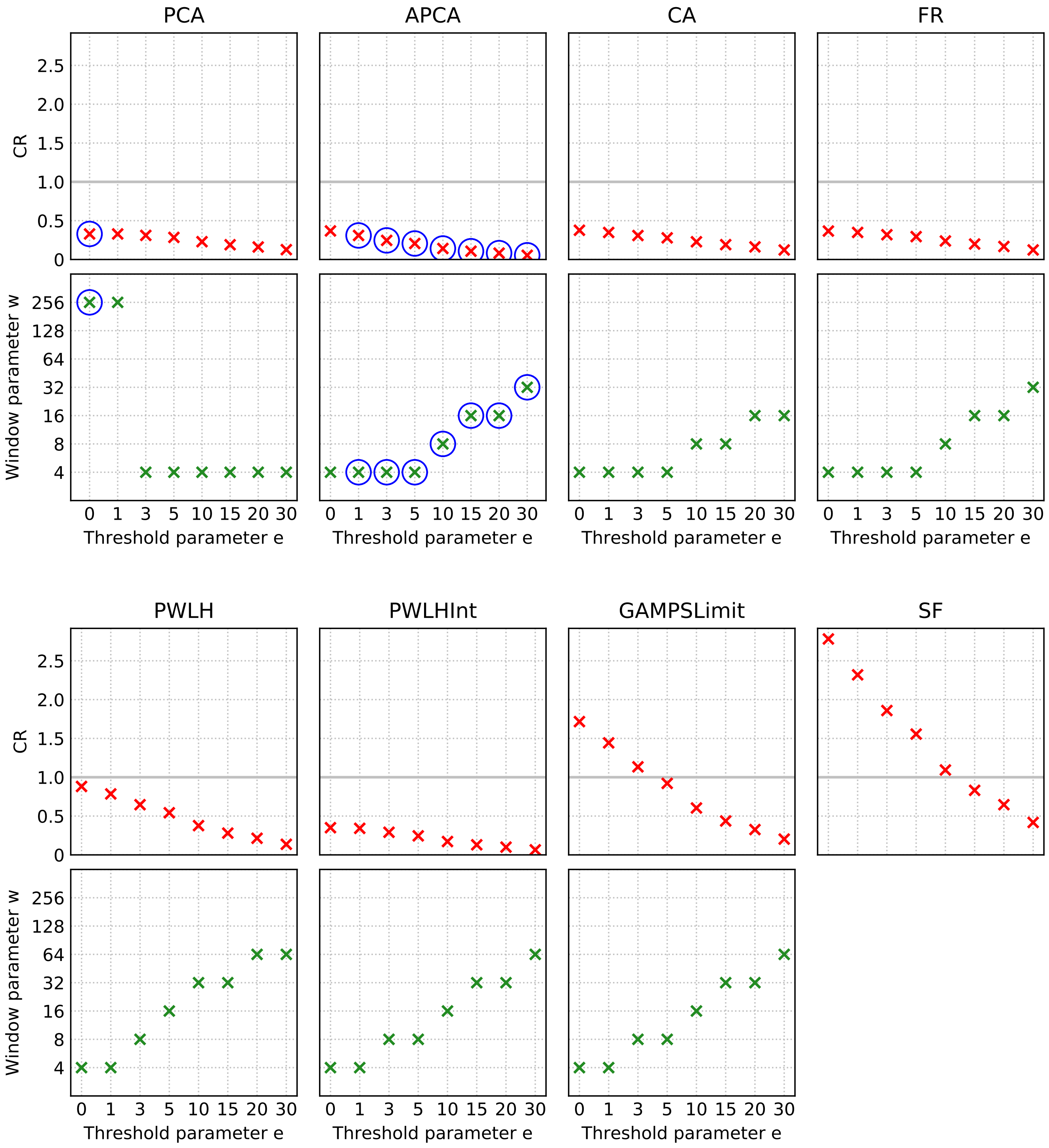

Figure 3.5: CR and window size parameter plots for every algorithm, for the data type "SST" of the dataset ElNino. For each threshold parameter $e \in E$, we use blue circles to highlight the markers for the minimum CR value and the best window size parameter (in the plots corresponding to the best algorithm)

PCA	APCA	FR

		e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e = 20		e = 30	
Dataset	Data Type	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w
IRKIS	VWC	0.2	4	0.18	4	0.12	5	0.07	6	0.03	7	0.02	8	0.02	8	0.01	8
SST	SST	0.61	8	0.28	3	0.14	5	0.09	6	0.05	7	0.03	8	0.02	8	0.02	8
ADCP	Vel	0.68	8	0.68	8	0.67	2	0.61	2	0.48	2	0.41	2	0.35	3	0.26	3
Solar	GHI	0.78	2	0.76	3	0.71	4	0.67	4	0.59	4	0.52	4	0.47	4	0.38	4
	DNI	0.76	2	0.72	4	0.66	4	0.61	4	0.54	4	0.49	4	0.43	4	0.36	4
	DHI	0.78	2	0.77	2	0.72	4	0.68	4	0.6	4	0.54	4	0.48	4	0.39	4
ElNino	Lat	0.16	4	0.16	4	0.16	4	0.15	4	0.12	4	0.1	5	0.09	5	0.06	6
	Long	0.17	3	0.17	4	0.13	4	0.12	5	0.09	6	0.07	6	0.05	7	0.02	8
	Z. Wind	0.31	8	0.31	8	0.31	8	0.31	8	0.27	2	0.24	2	0.21	2	0.16	3
	M. Wind	0.31	8	0.31	8	0.31	8	0.31	8	0.29	2	0.26	2	0.23	2	0.19	2
	Humidity	0.23	8	0.23	8	0.23	8	0.23	8	0.21	2	0.18	2	0.16	2	0.13	2
	AirTemp	0.33	8	0.33	8	0.3	2	0.27	2	0.22	2	0.19	3	0.17	3	0.13	4
	SST	0.33	8	0.31	2	0.25	2	0.21	2	0.14	3	0.11	4	0.08	4	0.05	5
Hail	Lat	1.0	8	1.0	8	0.9	2	0.83	2	0.71	2	0.65	3	0.57	3	0.47	3
	Long	1.0	8	1.0	8	0.86	2	0.78	2	0.65	2	0.55	3	0.49	3	0.39	4
	Size	0.81	2	0.81	2	0.81	2	0.81	2	0.81	2	0.81	2	0.81	2	0.64	3
Tornado	Lat	1.0	8	0.85	2	0.71	2	0.65	2	0.54	3	0.47	3	0.42	4	0.33	4
	Long	1.0	8	0.82	2	0.65	2	0.58	3	0.46	3	0.4	4	0.35	4	0.28	4
Wind	Lat	1.0	8	1.0	8	0.89	2	0.81	2	0.7	2	0.62	3	0.56	3	0.47	3
	Long	1.0	8	0.95	2	0.8	2	0.73	2	0.62	3	0.54	3	0.49	3	0.4	4
	Speed	0.65	4	0.44	3	0.26	6	0.17	7	0.16	5	0.12	6	0.1	6	0.08	6

TABLA 3.3: Compression performance results for each data type of each dataset, regarding the best CAI obtained for each threshold parameter. Each row contains information pertaining a certain data type. For each threshold, the first column shows the minimum CR, the second column shows $\log_2$ of the best window size parameter, and the best algorithm is represented by a certain cell color. We use red font to mark the cases in which the minimum CR is greater than 100%.

Table 3.6 summarizes the compression performance results for each data type of each dataset. Each row contains information relative to certain data type. For example, the 13th row shows summarized results for the data type "SST" of the dataset ElNino, which were presented in more detail in Figure 3.5. For each threshold, the first column shows the best (minimum) CR, and the second column shows the base-2 logarithm of the OWS for the best algorithm (the one that achieves this CR). The best algorithm itself is represented by a cell color code. We use red font to mark the cases in which the minimum CR is greater than 1. In these cases, the best algorithm has a compression performance that is inferior to that of the trivial CAI, *Base*.

We observe that there are three algorithms (PCA, APCA, and FR) that are best for at least one of the 168 possible data type and threshold parameter combinations. APCA is the best algorithm in exactly 134 combinations (80%), including every case in which $e \geq 10$, and most of the cases in which $e \in [1, 3, 5]$. PCA is the best algorithm in 31 combinations (18%), including most of the lossless cases, while FR is the best algorithm in only 3 combinations (2%), all of them for data type "Speed" of the dataset Wind.

TODO: Analize the RD between PCA and APCA (and FR?). Table with data in next page.

PCA	APCA	FR	PWLHInt

		e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e = 20		e = 30	
Dataset	Data Type	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w
IRKIS	VWC	20.9	5	18.67	5	12.71	6	6.9	7	3.09	8	2.29	7	1.84	7	1.41	7
SST	SST	60.85	7	28.44	4	13.95	4	8.96	5	4.74	6	3.16	7	2.48	7	1.89	7
ADCP	Vel	68.24	7	68.24	7	68.22	8	65.23	3	50.8	3	41.49	3	35.9	2	26.27	4
Solar	GHI	81.0	4	76.14	4	71.7	3	67.8	3	59.74	3	54.07	3	49.08	3	39.84	5
	DNI	76.4	4	72.48	3	66.48	3	62.44	3	55.6	3	50.57	3	45.45	5	36.93	5
	DHI	80.74	4	78.16	2	71.91	3	68.18	3	61.25	3	55.26	3	49.96	3	41.37	5
ElNino	Lat	16.12	3	16.12	3	16.01	3	15.42	3	12.43	5	10.08	6	8.66	6	5.82	7
	Long	17.39	4	17.08	3	13.12	5	11.82	4	8.66	5	6.62	7	4.96	6	2.4	7
	Zonal Winds	31.47	7	31.47	7	31.47	7	31.47	7	29.65	3	25.19	3	21.6	3	16.56	2
	Merid. Winds	31.47	7	31.47	7	31.47	7	31.47	7	31.46	8	27.91	3	24.95	3	19.8	3
	Humidity	23.11	7	23.11	7	23.11	7	23.11	7	22.1	3	19.4	3	16.89	3	13.05	3
	AirTemp	32.69	7	32.69	7	32.34	3	28.97	3	22.97	3	19.51	2	17.24	4	13.42	3
	SST	32.92	7	32.91	8	25.74	3	20.91	3	14.52	4	10.92	3	8.35	5	5.56	6
Hail	Lat	100.07	7	100.07	7	95.57	3	87.08	3	73.35	3	64.91	2	59.08	2	47.11	4
	Long	100.06	7	100.06	7	90.77	3	80.87	3	65.24	3	57.24	4	49.35	4	39.65	3
	Size	83.97	3	83.95	3	83.95	3	83.94	3	83.91	3	83.88	3	83.87	3	64.6	2
Tornado	Lat	100.08	7	90.94	3	73.43	3	65.95	3	56.33	2	47.57	4	42.13	3	34.54	5
	Long	100.14	7	86.92	3	65.99	3	58.84	2	46.42	4	40.26	3	36.09	5	28.78	5
Wind	Lat	100.06	7	100.06	7	93.92	3	85.1	3	70.98	3	63.08	2	57.92	4	47.53	4
	Long	100.06	7	100.03	8	83.85	3	75.03	3	63.08	2	55.61	4	48.84	4	40.81	5
	Speed	65.76	3	45.45	4	26.36	7	16.82	6	16.1	6	12.39	5	10.63	5	8.39	7

PCA	APCA	FR	PWLHInt	PWLH	CA

Dataset	Data Type	e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e = 20		e = 30	
		CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w
IRKIS	VWC	27.81	5	27.81	5	21.21	5	12.99	6	6.8	7	4.97	8	3.44	8	1.63	8
SST	SST	63.83	2	33.37	4	15.39	6	9.73	7	4.92	8	3.33	8	2.63	8	2.03	8
ADCP	Vel	73.3	2	72.88	2	68.22	8	66.83	2	59.28	3	51.1	3	44.7	3	34.02	4
Solar	GHI	81.0	4	76.51	2	75.65	2	73.26	4	66.97	4	62.83	4	59.88	4	54.06	4
	DNI	76.4	4	74.29	4	72.55	4	70.86	4	67.21	4	64.23	4	61.64	4	57.22	4
	DHI	80.74	4	78.16	2	75.57	4	72.78	4	67.69	4	63.49	4	60.08	4	54.61	4
ElNino	Lat	22.54	4	22.54	4	22.54	4	22.54	4	20.79	2	18.06	2	15.94	3	12.17	3
	Long	24.69	4	24.48	4	21.77	4	19.99	4	15.31	5	12.43	6	10.02	6	5.8	8
	Zonal Winds	34.51	2	34.51	2	33.25	2	31.56	2	31.12	2	28.38	2	26.04	3	20.98	3
	Merid. Winds	34.57	2	34.57	2	34.1	2	33.16	2	31.46	8	30.04	2	28.33	2	24.5	3
	Humidity	25.12	2	25.12	2	24.98	2	23.42	2	22.38	2	20.88	2	19.34	3	15.85	3
	AirTemp	34.95	2	34.71	2	32.44	2	30.31	2	25.93	3	22.76	4	20.04	4	16.43	5
	SST	35.01	2	32.91	8	29.13	3	24.5	3	17.19	4	13.0	5	10.0	5	6.5	6
Hail	Lat	108.59	2	102.05	2	99.1	2	95.09	2	86.27	3	79.57	3	73.8	3	63.72	4
	Long	107.97	2	100.96	2	95.88	2	90.95	3	79.94	3	72.14	3	65.42	4	55.75	4
	Size	94.07	2	94.05	2	94.05	2	94.05	2	94.03	2	94.02	2	92.76	2	85.3	3
Tornado	Lat	108.2	2	95.5	2	84.04	3	76.63	3	66.4	4	58.48	4	53.72	4	45.59	4
	Long	107.67	2	90.47	2	74.82	3	68.49	3	55.2	4	48.66	4	44.37	5	37.56	5
Wind	Lat	108.36	2	101.6	2	99.29	2	96.6	2	89.73	2	83.81	2	79.46	3	71.4	2
	Long	107.78	2	100.03	8	95.67	2	91.45	2	83.21	2	77.38	2	72.14	2	64.03	2
	Speed	68.31	3	58.97	4	28.02	4	23.98	4	16.51	7	16.32	7	16.12	7	15.78	7

		e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e = 20		e = 30	
Dataset	Data Type	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w
IRKIS	VWC	34.15	2	31.79	2	25.05	2	16.87	3	10.87	3	8.99	4	7.49	4	6.15	4
SST	SST	60.84	8	41.79	2	26.98	2	22.26	2	14.61	3	11.91	3	10.39	4	8.03	4
ADCP	Vel	68.22	8	68.22	8	68.22	8	68.22	8	65.08	2	59.58	2	53.76	2	43.52	2
Solar	GHI	77.65	2	76.51	2	75.65	2	74.95	2	72.61	2	70.42	2	68.26	2	64.0	2
	DNI	75.93	2	74.35	2	72.75	2	71.49	2	69.79	2	68.18	2	66.14	2	62.13	2
	DHI	77.66	2	77.43	2	76.11	2	75.33	2	73.98	2	72.64	2	70.74	2	66.68	2
ElNino	Lat	25.05	2	25.05	2	24.89	2	24.09	2	20.79	2	18.06	2	15.94	3	12.17	3
	Long	27.24	2	26.83	2	21.87	2	20.44	2	16.42	3	13.59	3	11.33	3	8.13	3
	Zonal Winds	31.46	8	31.46	8	31.46	8	31.46	8	31.46	8	30.82	2	29.48	2	25.92	2
	Merid. Winds	31.46	8	31.46	8	31.46	8	31.46	8	31.46	8	31.38	2	30.6	2	28.19	2
	Humidity	23.1	8	23.1	8	23.1	8	23.1	8	23.1	8	22.91	2	22.09	2	19.88	2
	AirTemp	32.68	8	32.68	8	32.68	8	31.93	2	28.83	2	25.89	2	23.73	2	20.67	2
	SST	32.91	8	32.91	8	31.05	2	28.46	2	22.91	2	18.98	2	16.1	2	12.68	2
Hail	Lat	100.04	8	100.04	8	99.52	2	97.25	2	91.04	2	85.63	2	80.74	2	71.26	2
	Long	100.03	8	100.03	8	98.88	2	95.54	2	86.75	2	79.54	2	73.26	2	64.17	2
	Size	94.07	2	94.05	2	94.05	2	94.05	2	94.03	2	94.02	2	94.02	2	87.23	2
Tornado	Lat	100.05	8	99.76	2	93.9	2	89.21	2	79.92	2	73.98	2	69.53	2	61.64	2
	Long	100.11	8	98.88	2	89.2	2	83.18	2	73.42	2	67.64	2	62.02	2	54.99	2
Wind	Lat	100.03	8	100.03	8	99.29	2	96.6	2	89.73	2	83.81	2	79.48	2	71.4	2
	Long	100.03	8	100.03	8	96.07	2	91.45	2	83.21	2	77.38	2	72.14	2	64.03	2
	Speed	100.04	8	67.73	2	55.23	2	44.21	2	37.59	2	35.05	3	32.79	3	30.46	3

		e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e = 20		e = 30	
Dataset	Data Type	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w
IRKIS	VWC	20.32	4	18.35	4	12.37	5	6.77	6	3.07	7	2.22	8	1.71	8	1.21	8
SST	SST	66.1	2	28.12	3	13.64	5	8.88	6	4.63	7	3.15	8	2.39	8	1.72	8
ADCP	Vel	77.52	2	74.51	2	66.8	2	61.07	2	48.44	2	40.9	2	34.9	3	25.93	3
Solar	GHI	82.64	2	76.1	3	71.39	4	67.2	4	58.52	4	52.41	4	47.03	4	37.78	4
	DNI	78.91	2	72.22	4	65.75	4	61.37	4	53.98	4	48.55	4	43.36	4	35.66	4
	DHI	82.07	2	78.16	2	71.62	4	67.6	4	60.12	4	53.62	4	47.86	4	38.71	4
ElNino	Lat	15.96	4	15.96	4	15.82	4	15.11	4	12.34	4	9.89	5	8.61	5	5.76	6
	Long	17.36	3	17.05	4	13.04	4	11.75	5	8.65	6	6.56	6	4.93	7	2.37	8
	Zonal Winds	37.11	2	37.11	2	33.25	2	31.56	2	27.36	2	23.5	2	20.54	2	16.44	3
	Merid. Winds	37.29	2	37.29	2	34.1	2	33.16	2	29.16	2	25.86	2	23.33	2	19.15	2
	Humidity	26.39	2	26.29	2	25.38	2	23.42	2	20.51	2	18.14	2	16.01	2	12.94	2
	AirTemp	36.2	2	34.96	2	30.33	2	27.39	2	22.42	2	19.24	3	16.76	3	13.31	4
	SST	36.79	2	30.96	2	24.6	2	20.61	2	14.17	3	10.66	4	8.21	4	5.42	5
Hail	Lat	114.81	2	102.05	2	89.83	2	82.62	2	71.49	2	64.62	3	57.49	3	46.75	3
	Long	114.14	2	100.96	2	85.91	2	77.5	2	65.06	2	55.38	3	48.72	3	38.74	4
	Size	80.61	2	80.59	2	80.59	2	80.58	2	80.56	2	80.53	2	80.52	2	64.35	3
Tornado	Lat	111.97	2	85.43	2	70.63	2	65.17	2	54.17	3	46.78	3	41.95	4	33.48	4
	Long	111.05	2	82.12	2	65.09	2	57.66	3	45.55	3	39.88	4	34.84	4	28.41	4
Wind	Lat	113.34	2	101.6	2	88.74	2	81.29	2	69.82	2	62.44	3	56.18	3	47.15	3
	Long	112.6	2	95.41	2	80.29	2	73.21	2	62.06	3	54.33	3	48.52	3	39.73	4
	Speed	98.1	2	43.82	3	28.02	4	23.98	4	15.71	5	12.29	6	10.33	6	8.21	6

PCA	APCA

		e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e = 20		e = 30	
Dataset	Data Type	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD
IRKIS	VWC	20.32	40.52	18.35	42.28	12.37	50.62	6.77	59.86	3.07	71.73	2.22	75.33	1.71	77.21	1.21	80.28
SST	SST	60.84	-8.64	28.12	32.71	13.64	49.42	8.88	60.11	4.63	68.29	3.15	73.53	2.39	76.95	1.72	78.54
ADCP	Vel	68.22	-13.62	68.22	-9.22	66.8	2.08	61.07	10.48	48.44	25.57	40.9	31.35	34.9	35.08	25.93	40.41
Solar	GHI	77.65	-6.42	76.1	0.53	71.39	5.62	67.2	10.34	58.52	19.4	52.41	25.58	47.03	31.11	37.78	40.97
	DNI	75.93	-3.92	72.22	2.86	65.75	9.62	61.37	14.16	53.98	22.65	48.55	28.79	43.36	34.44	35.66	42.61
	DHI	77.66	-5.68	77.43	-0.94	71.62	5.9	67.6	10.26	60.12	18.74	53.62	26.18	47.86	32.34	38.71	41.94
ElNino	Lat	15.96	36.3	15.96	36.3	15.82	36.44	15.11	37.28	12.34	40.65	9.89	45.2	8.61	45.95	5.76	52.65
	Long	17.36	36.28	17.05	36.44	13.04	40.39	11.75	42.54	8.65	47.32	6.56	51.76	4.93	56.53	2.37	70.89
	Zonal Winds	31.46	-17.98	31.46	-17.98	31.46	-5.69	31.46	-0.33	27.36	13.02	23.5	23.75	20.54	30.33	16.44	36.58
	Merid. Winds	31.46	-18.54	31.46	-18.54	31.46	-8.42	31.46	-5.4	29.16	7.32	25.86	17.59	23.33	23.75	19.15	32.07
	Humidity	23.1	-14.26	23.1	-13.8	23.1	-9.89	23.1	-1.37	20.51	11.21	18.14	20.82	16.01	27.55	12.94	34.9
	AirTemp	32.68	-10.78	32.68	-6.97	30.33	7.2	27.39	14.24	22.42	22.22	19.24	25.69	16.76	29.38	13.31	35.61
	SST	32.91	-11.79	30.96	5.91	24.6	20.8	20.61	27.61	14.17	38.16	10.66	43.82	8.21	49.02	5.42	57.25
Hail	Lat	100.04	-14.77	100.04	-2.01	89.83	9.73	82.62	15.05	71.49	21.47	64.62	24.54	57.49	28.8	46.75	34.4
	Long	100.03	-14.11	100.03	-0.93	85.91	13.12	77.5	18.88	65.06	25.0	55.38	30.37	48.72	33.5	38.74	39.63
	Size	80.61	14.31	80.59	14.32	80.59	14.32	80.58	14.32	80.56	14.33	80.53	14.35	80.52	14.35	64.35	26.22
Tornado	Lat	100.05	-11.92	85.43	14.36	70.63	24.78	65.17	26.95	54.17	32.22	46.78	36.77	41.95	39.67	33.48	45.69
	Long	100.11	-10.92	82.12	16.95	65.09	27.03	57.66	30.68	45.55	37.96	39.88	41.04	34.84	43.82	28.41	48.34
Wind	Lat	100.03	-13.31	100.03	-1.57	88.74	10.63	81.29	15.85	69.82	22.19	62.44	25.5	56.18	29.32	47.15	33.96
	Long	100.03	-12.56	95.41	4.62	80.29	16.42	73.21	19.94	62.06	25.42	54.33	29.79	48.52	32.74	39.73	37.96
	Speed	98.1	1.94	43.82	35.31	28.02	49.27	23.98	45.76	15.71	58.2	12.29	64.93	10.33	68.48	8.21	73.05

PCA	APCA	FR

		e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e =
Dataset	Data Type	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD	CR
IRKIS	VWC	0.28	26.94	0.28	34.0	0.25	49.94	0.22	68.95	0.13	76.68	0.07	67.29	0.03
SST	SST	0.64	4.69	0.33	15.74	0.15	11.32	0.1	8.69	0.05	5.74	0.03	5.23	0.03
ADCP	Vel	0.73	6.92	0.73	6.39	0.7	4.34	0.67	8.61	0.59	18.28	0.51	19.95	0.45
Solar	GHI	0.81	4.13	0.78	2.77	0.76	5.78	0.73	8.27	0.67	12.62	0.63	16.59	0.6
	DNI	0.76	0.61	0.74	2.79	0.73	9.37	0.71	13.39	0.67	19.68	0.64	24.42	0.62
	DHI	0.81	3.82	0.81	4.1	0.76	5.22	0.73	7.11	0.68	11.19	0.63	15.55	0.6
ElNino	Lat	0.23	29.23	0.23	29.23	0.23	29.83	0.23	33.0	0.22	44.82	0.21	53.84	0.2
	Long	0.25	29.72	0.24	30.34	0.24	46.18	0.22	47.63	0.18	51.24	0.16	58.83	0.14
	Z. Wind	0.35	8.85	0.35	8.85	0.35	8.85	0.34	6.81	0.31	12.08	0.28	17.18	0.26
	M. Wind	0.35	9.0	0.35	9.0	0.35	9.0	0.34	8.32	0.32	9.27	0.3	13.92	0.28
	Humidity	0.25	8.03	0.25	8.03	0.25	7.53	0.24	4.83	0.22	8.35	0.21	13.1	0.19
	AirTemp	0.35	6.48	0.35	5.85	0.32	6.52	0.3	9.63	0.26	13.52	0.23	15.48	0.2
	SST	0.35	6.0	0.34	9.42	0.29	15.56	0.24	15.88	0.17	17.57	0.13	17.94	0.1
Hail	Lat	1.09	7.88	1.04	4.23	0.99	9.35	0.95	13.11	0.86	17.13	0.8	18.79	0.74
	Long	1.08	7.35	1.03	2.89	0.96	10.39	0.91	14.78	0.8	18.61	0.72	23.22	0.65
	Size	1.0	19.31	1.0	19.33	0.99	18.94	0.99	18.39	0.97	17.07	0.94	14.71	0.93
Tornado	Lat	1.08	7.54	0.95	10.55	0.84	15.95	0.77	14.96	0.66	18.43	0.58	20.01	0.54
	Long	1.08	7.02	0.9	9.23	0.75	13.01	0.68	15.81	0.55	17.48	0.49	18.04	0.44
Wind	Lat	1.08	7.69	1.05	4.92	1.0	11.36	0.97	15.86	0.9	22.49	0.84	25.71	0.79
	Long	1.08	7.19	1.02	6.41	0.96	16.07	0.92	20.15	0.83	25.56	0.78	30.33	0.74
	Speed	0.68	4.14	0.62	29.0	0.41	37.17	0.41	59.17	0.41	61.68	0.41	69.96	0.41

Tabla 3.4: CoderPWLHInt vs. BEST

PCA	APCA	FR

		e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e = 20	
Dataset	Data Type	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD
IRKIS	VWC	0.2	0	0.18	0	0.12	0	0.07	0	0.03	0	0.02	0	0.02	0
SST	SST	0.66	7.96	0.28	0	0.14	0	0.09	0	0.05	0	0.03	0	0.02	0
ADCP	Vel	0.78	11.99	0.75	8.44	0.67	0	0.61	0	0.48	0	0.41	0	0.35	0
Solar	GHI	0.83	6.03	0.76	0	0.71	0	0.67	0	0.59	0	0.52	0	0.47	0
	DNI	0.79	3.78	0.72	0	0.66	0	0.61	0	0.54	0	0.49	0	0.43	0
	DHI	0.82	5.37	0.78	0.94	0.72	0	0.68	0	0.6	0	0.54	0	0.48	0
ElNino	Lat	0.16	0	0.16	0	0.16	0	0.15	0	0.12	0	0.1	0	0.09	0
	Long	0.17	0	0.17	0	0.13	0	0.12	0	0.09	0	0.07	0	0.05	0
	Z. Wind	0.37	15.24	0.37	15.24	0.33	5.38	0.32	0.33	0.27	0	0.24	0	0.21	0
	M. Wind	0.37	15.64	0.37	15.64	0.34	7.76	0.33	5.13	0.29	0	0.26	0	0.23	0
	Humidity	0.26	12.48	0.26	12.13	0.25	9.0	0.23	1.35	0.21	0	0.18	0	0.16	0
	AirTemp	0.36	9.73	0.35	6.51	0.3	0	0.27	0	0.22	0	0.19	0	0.17	0
	SST	0.37	10.55	0.31	0	0.25	0	0.21	0	0.14	0	0.11	0	0.08	0
Hail	Lat	1.15	12.87	1.02	1.97	0.9	0	0.83	0	0.71	0	0.65	0	0.57	0
	Long	1.14	12.36	1.01	0.92	0.86	0	0.78	0	0.65	0	0.55	0	0.49	0
	Size	0.81	0	0.81	0	0.81	0	0.81	0	0.81	0	0.81	0	0.81	0
Tornado	Lat	1.12	10.65	0.85	0	0.71	0	0.65	0	0.54	0	0.47	0	0.42	0
	Long	1.11	9.85	0.82	0	0.65	0	0.58	0	0.46	0	0.4	0	0.35	0
Wind	Lat	1.13	11.74	1.02	1.55	0.89	0	0.81	0	0.7	0	0.62	0	0.56	0
	Long	1.13	11.16	0.95	0	0.8	0	0.73	0	0.62	0	0.54	0	0.49	0
	Speed	0.98	33.25	0.44	0	0.28	7.57	0.24	29.96	0.16	0	0.12	0	0.1	0

Tabla 3.5: CoderAPCA vs. BEST

PCA	APCA	FR

		e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e =
Dataset	Data Type	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD	CR	RD	CR
IRKIS	VWC	0.34	40.52	0.32	42.28	0.25	50.62	0.17	59.86	0.11	71.73	0.09	75.33	0.07
SST	SST	0.61	0	0.42	32.71	0.27	49.42	0.22	60.11	0.15	68.29	0.12	73.53	0.1
ADCP	Vel	0.68	0	0.68	0	0.68	2.08	0.68	10.48	0.65	25.57	0.6	31.35	0.54
Solar	GHI	0.78	0	0.77	0.53	0.76	5.62	0.75	10.34	0.73	19.4	0.7	25.58	0.68
	DNI	0.76	0	0.74	2.86	0.73	9.62	0.71	14.16	0.7	22.65	0.68	28.79	0.66
	DHI	0.78	0	0.77	0	0.76	5.9	0.75	10.26	0.74	18.74	0.73	26.18	0.71
ElNino	Lat	0.25	36.3	0.25	36.3	0.25	36.44	0.24	37.28	0.21	40.65	0.18	45.2	0.16
	Long	0.27	36.28	0.27	36.44	0.22	40.39	0.2	42.54	0.16	47.32	0.14	51.76	0.11
	Z. Wind	0.31	0	0.31	0	0.31	0	0.31	0	0.31	13.02	0.31	23.75	0.29
	M. Wind	0.31	0	0.31	0	0.31	0	0.31	0	0.31	7.32	0.31	17.59	0.31
	Humidity	0.23	0	0.23	0	0.23	0	0.23	0	0.23	11.21	0.23	20.82	0.22
	AirTemp	0.33	0	0.33	0	0.33	7.2	0.32	14.24	0.29	22.22	0.26	25.69	0.24
	SST	0.33	0	0.33	5.91	0.31	20.8	0.28	27.61	0.23	38.16	0.19	43.82	0.16
Hail	Lat	1.0	0	1.0	0	1.0	9.73	0.97	15.05	0.91	21.47	0.86	24.54	0.81
	Long	1.0	0	1.0	0	0.99	13.12	0.96	18.88	0.87	25.0	0.8	30.37	0.73
	Size	0.94	14.31	0.94	14.32	0.94	14.32	0.94	14.32	0.94	14.33	0.94	14.35	0.94
Tornado	Lat	1.0	0	1.0	14.36	0.94	24.78	0.89	26.95	0.8	32.22	0.74	36.77	0.7
	Long	1.0	0	0.99	16.95	0.89	27.03	0.83	30.68	0.73	37.96	0.68	41.04	0.62
Wind	Lat	1.0	0	1.0	0	0.99	10.63	0.97	15.85	0.9	22.19	0.84	25.5	0.79
	Long	1.0	0	1.0	4.62	0.96	16.42	0.91	19.94	0.83	25.42	0.77	29.79	0.72
	Speed	1.0	34.54	0.68	35.31	0.55	53.11	0.44	62.01	0.38	58.2	0.35	64.93	0.33

Tabla 3.6: CoderPCA vs. BEST

TODO: Comparison with the algorithm gzip. Table with data in next page.

GZIP	PCA	APCA	FR

		e = 0		e = 1		e = 3		e = 5		e = 10		e = 15		e = 20		e = 30	
Dataset	Data Type	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w	CR	w
IRKIS	VWC	13.44		13.44		12.37	5	6.77	6	3.07	7	2.22	8	1.71	8	1.21	8
SST	SST	52.06		28.12	3	13.64	5	8.88	6	4.63	7	3.15	8	2.39	8	1.72	8
ADCP	Vel	61.38		61.38		61.38		61.07	2	48.44	2	40.9	2	34.9	3	25.93	3
Solar	GHI	69.01		69.01		69.01		67.2	4	58.52	4	52.41	4	47.03	4	37.78	4
	DNI	66.88		66.88		65.75	4	61.37	4	53.98	4	48.55	4	43.36	4	35.66	4
	DHI	61.01		61.01		61.01		61.01		60.12	4	53.62	4	47.86	4	38.71	4
ElNino	Lat	7.89		7.89		7.89		7.89		7.89		7.89		7.89		5.76	6
	Long	7.1		7.1		7.1		7.1		7.1		6.56	6	4.93	7	2.37	8
	Zonal Winds	31.46	8	31.46	8	31.46	8	31.46	8	27.36	2	23.5	2	20.54	2	16.44	3
	Merid. Winds	31.46	8	31.46	8	31.46	8	31.46	8	29.16	2	25.86	2	23.33	2	19.15	2
	Humidity	23.1	8	23.1	8	23.1	8	23.1	8	20.51	2	18.14	2	16.01	2	12.94	2
	AirTemp	32.68	8	32.68	8	30.33	2	27.39	2	22.42	2	19.24	3	16.76	3	13.31	4
	SST	32.43		30.96	2	24.6	2	20.61	2	14.17	3	10.66	4	8.21	4	5.42	5
Hail	Lat	100.04	8	100.04	8	89.83	2	82.62	2	71.49	2	64.62	3	57.49	3	46.75	3
	Long	100.03	8	100.03	8	85.91	2	77.5	2	65.06	2	55.38	3	48.72	3	38.74	4
	Size	36.73		36.73		36.73		36.73		36.73		36.73		36.73		36.73	
Tornado	Lat	100.05	8	85.43	2	70.63	2	65.17	2	54.17	3	46.78	3	41.95	4	33.48	4
	Long	100.11	8	82.12	2	65.09	2	57.66	3	45.55	3	39.88	4	34.84	4	28.41	4
Wind	Lat	100.03	8	100.03	8	88.74	2	81.29	2	69.82	2	62.44	3	56.18	3	47.15	3
	Long	100.03	8	95.41	2	80.29	2	73.21	2	62.06	3	54.33	3	48.52	3	39.73	4
	Speed	65.49	4	43.82	3	25.9	6	16.79	7	15.71	5	12.29	6	10.33	6	8.21	6

TABLA 3.7: Compression performance results for each data type of each dataset, regarding the best CAI obtained for each threshold parameter. Each row contains information pertaining a certain data type. For each threshold, the first column shows the minimum CR, the second column shows $\log_2$ of the best window size parameter, and the best algorithm is represented by a certain cell color. We use red font to mark the cases in which the minimum CR is greater than 100%.

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)

- Poner las gráficas horizontales, 3 arriba y 3 abajo.

- Mencionar que CoderSlideFilter no tiene en cuenta el parámetro con el tamaño máximo de la ventana.

- Mencionar experimentos ventana local vs ventana global. (ver minuta de la reunión del lunes 10/06/2019).

TODO INFORME:

- 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 relación de compromiso entre el umbral y la tasa de compresión: al aumentar el umbral mejor la tasa de compresión (lógico).

- 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.