REPORT LAB EXERCISE: INTEGER CODING

Francesca Maria Palazzotto, CdLM Data, Algorithms, and Machine Intelligence

Course "Information Theory and Data Compression"

INTRODUCTION

This report described the statistics on the following experiments:

1. Number of bits required to encode 100 integers between 1 and 100 000 (consider integers 1, 1011, 2021, …).
2. Number of bits required to compress 100 random integers between 1 and 1000.
3. Number of bits required to encode a sequence of 1000 integers with a distribution chosen in advance.

On the third experiment, it will be considering 1000 odd integers to study the bits required and if there exists a property that characterizes the encoding of odd integers.

The codes considered are the following:

- Gamma code.
- Delta code.
- Fibonacci code.
- Levensthein code.

There will be a different section dedicated to Rice codes in which it will be shown the statistics of the number of bits used in the three different experiments based on the variation of the parameter k.

For each experiment, the statistical analysis is constructed in the following steps:

- Minimum and maximum to find errors, if any.
- Histogram and boxplot to have an idea about the distribution.
- Form indexes: symmetry and curtsies to establish if the distribution is like a Normal distribution.
- Positional indexes:
    - If the distribution is symmetric, determine average and standard deviation.
    - If the distribution is asymmetric, determine median and interquartile range.

In the file named *Ex.3_Integer_Coding.py*, there is all the analysis reported here, all the figures generated for each experiment and the implementation of the codes nominated above, both encoding and decoding, and a plot about another point of the exercise that is not reported here.

EXPERIMENT 1

The first step looks at the four different codes considered and how the number of bits used for each one change based on the number encoded.

It is possible to see how all the codes for the few first numbers, used the same number of bits, between 15 and 20, and, after that, bits start to increase rapidly, as we can see in Figure 1.

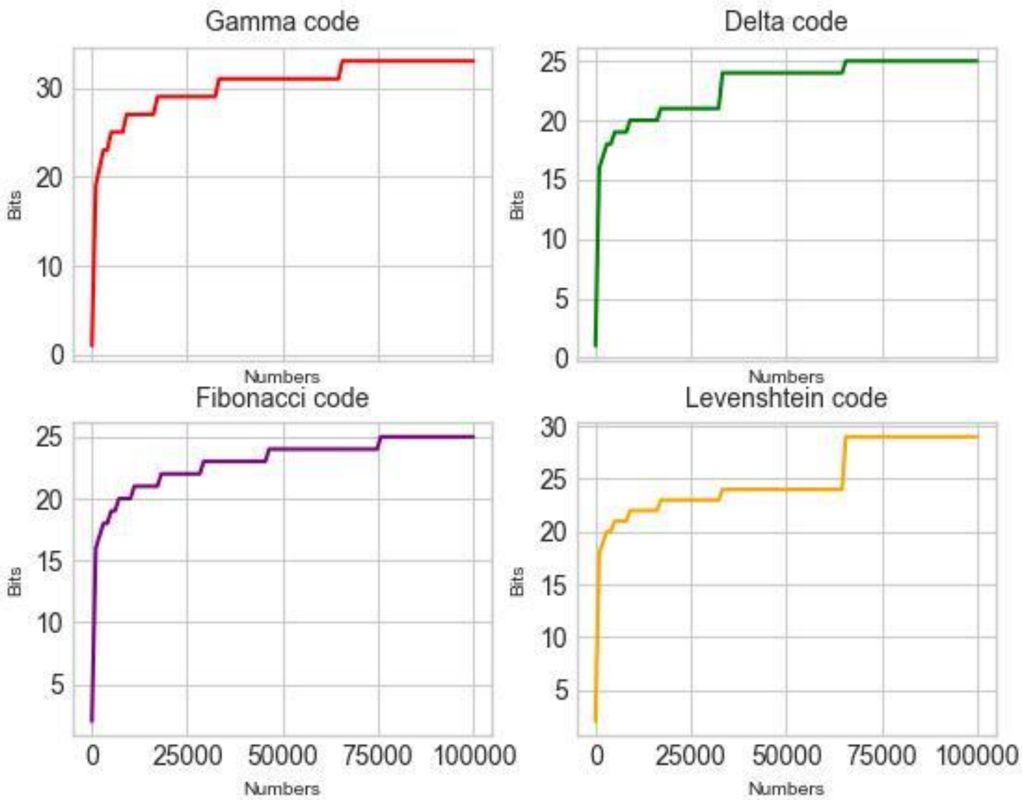

Figure 1: Plotting of codes.

As a second step, statistics of each code were studied. All the codes have an asymmetric distribution. Gamma and Delta codes share a minimum of 1 bit used, while Fibonacci and Levenshtein share a minimum of 2 bits used. Delta and Fibonacci codes share a maximum 25 bits used, while Gamma has a maximum of 33 bits and Levenshtein has a maximum of 29 bits.

The median is equal to 31 bits for Gamma code, while Delta, Fibonacci and Levenshtein codes share a median of 24 bits. The IQR is shared by Gamma and Delta codes, 4.00, Fibonacci code has 2.25 and Levenshtein code has a 6.00.

The following Figure 2 presents histograms and boxplots of each code. It is easy to see the asymmetric distribution of bits used of each code, the minimum, maximum, median and IQR of each code from the figures.

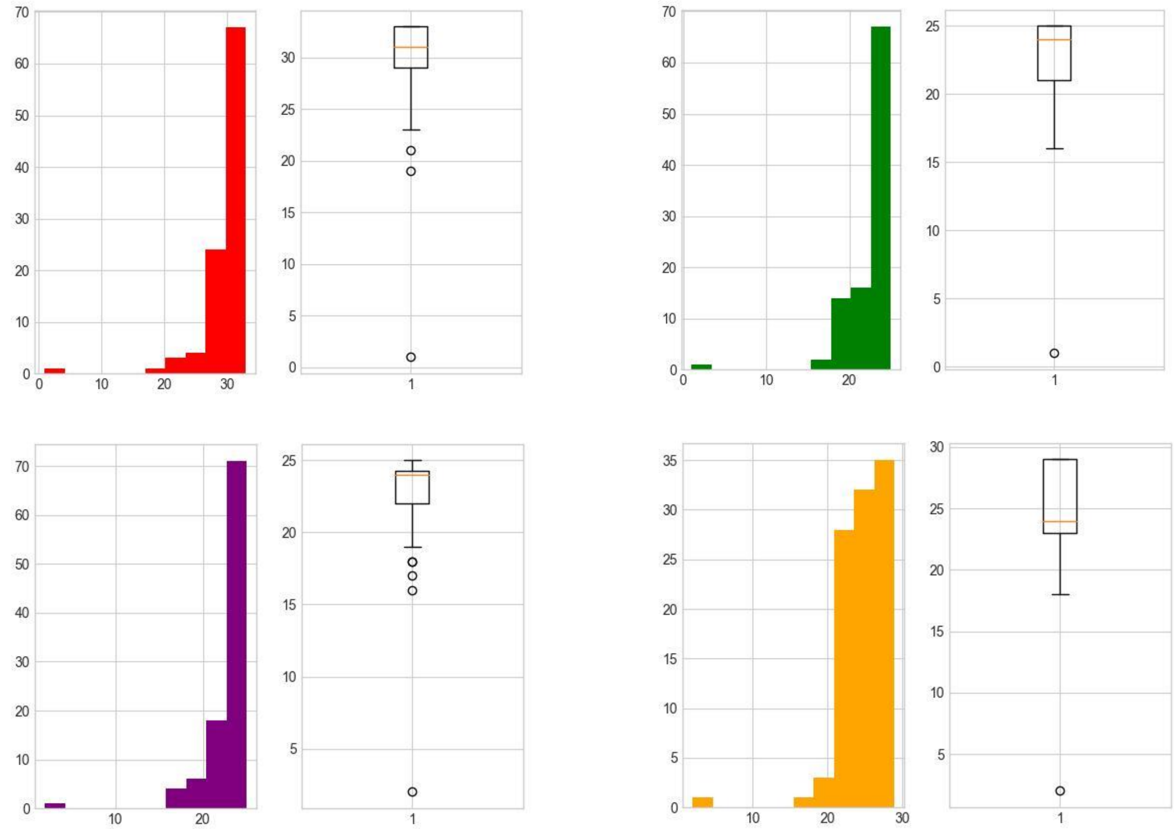

Figure 2: Gamma code (red), Delta code (green), Fibonacci code (purple), Levenshtein code (yellow)

EXPERIMENT 2

The first step looks at the four different codes considered and how the number of bits used for each one, changes based on the number encoded.

From Figure 3, it is possible to see how the codes use several bits in a range between 5 and 10 bits for the first number considered. There is a rapid increase of bits up to 250, after it continues to grow slowly, following the trend of a step function.

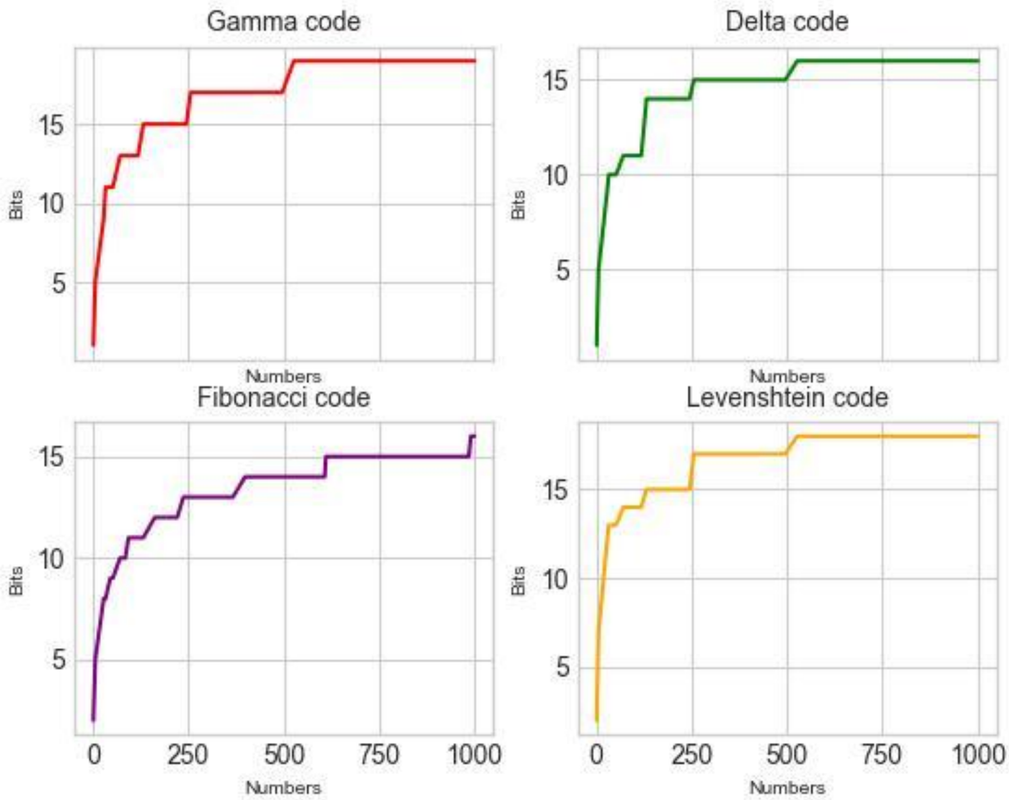

Figure 3: Plotting of bits used for each code.

Figure 4 presents the statistics studied during this experiment. The distributions of the four codes are asymmetric and dense on the right side of the distribution, which indicates that a greater number of bits are used, averaging between 10 and 15.

Gamma and Delta codes share a minimum of 1 bit used, while Fibonacci and Levensthein codes share a minimum of 2 bits. Delta ad Fibonacci codes share a maximum of 16 bits used, while Gamma has a maximum of 19 bits and Levensthein has a maximum of 18 bits used.

The median is equal to 17 bits for both Gamma and Levensthein codes, while it is equal to 15 bits and 14 bits respectively for Delta and Fibonacci codes. The IQR has value equal to 2.50, 1.25, 2.00, and 1.50 respectively for Gamma, Delta, Fibonacci and Levensthein code.

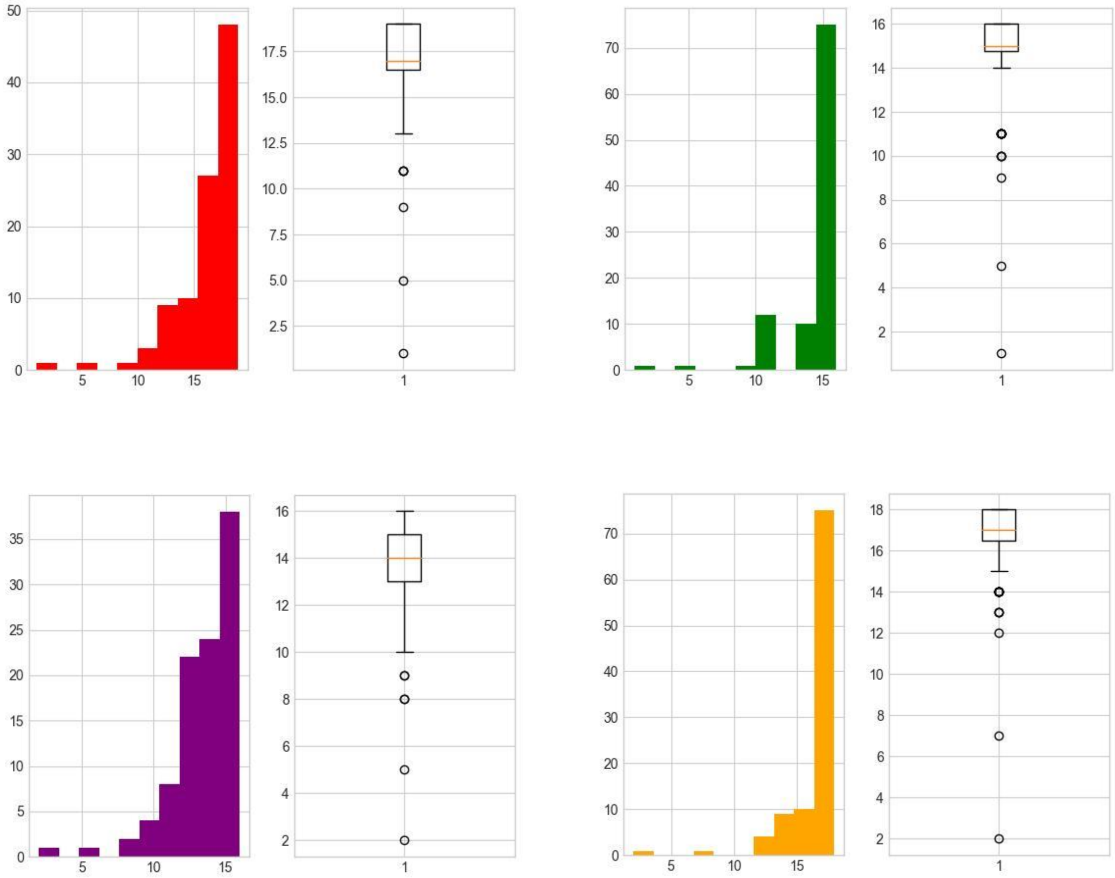

Figure 4: gamma code (red), delta code (green), Fibonacci code (purple), Levenshtein code (yellow)

In the figure above, it is possible to see histograms and boxplots of each code, in which there are visible all the characteristics described.

EXPERIMENT 3

The first step looks at the four different codes considered and how the number of bits used for each one, changes based on the number encoded.

Figure 5 shows the trend of the bits used by each code for the encoding of the odd numbers. It is possible to see how the number of bits used grows quickly, following the trend of a step function.

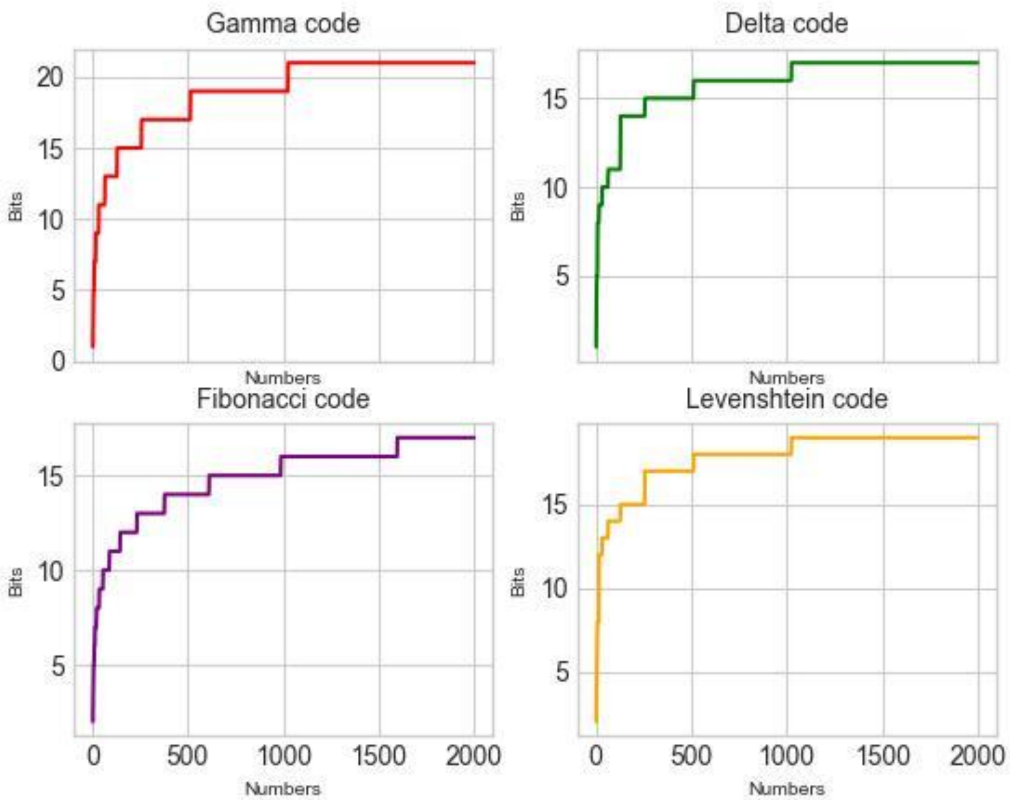

Figure 5: Plotting of bits used for each code.

Figure 6 presents the statistics studied during this last experiment. The distributions of the four codes are asymmetric and dense on the right side of the distribution, which indicates that a greater number of bits are used, averaging around 15.

Gamma and Delta codes share a minimum of 1 bit used, while Fibonacci and Levensthein codes share a minimum of 2 bits. Delta ad Fibonacci codes share a maximum of 17 bits used, while Gamma has a maximum of 21 bits and Levensthein has a maximum of 19 bits used.

The median is equal to 16 bits for both Delta and Levensthein codes, while it is equal to 19 bits and 18 bits respectively for Gamma and Levensthein codes. The IQR has value equal to 4 for Gamma code, while it is equal to 2 for Delta, Fibonacci and Levensthein codes. All these information is visible on Figure 6.

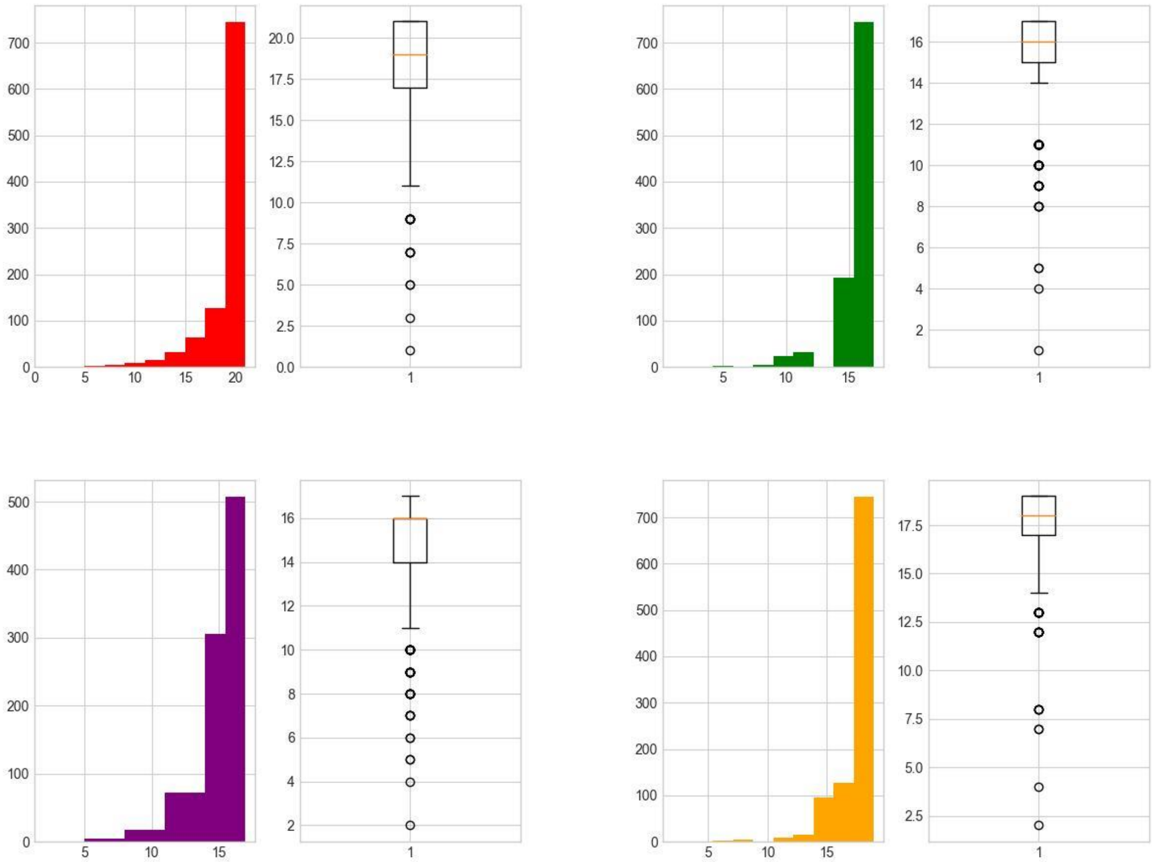

Figure 6: gamma code (red), delta code (green), Fibonacci code (purple), Levenshtein code (yellow)

EXPERIMENT 1 – 2 – 3: RICE CODES

This section presents an analysis of the application of Rice codes in the three different experiments proposed, using the different choice of the parameter *k*, which takes values from 1 to 10.

For each value of k, it was implemented the experiment and encoded the integers considered, and then it was evaluated the number of bits used for each of them.

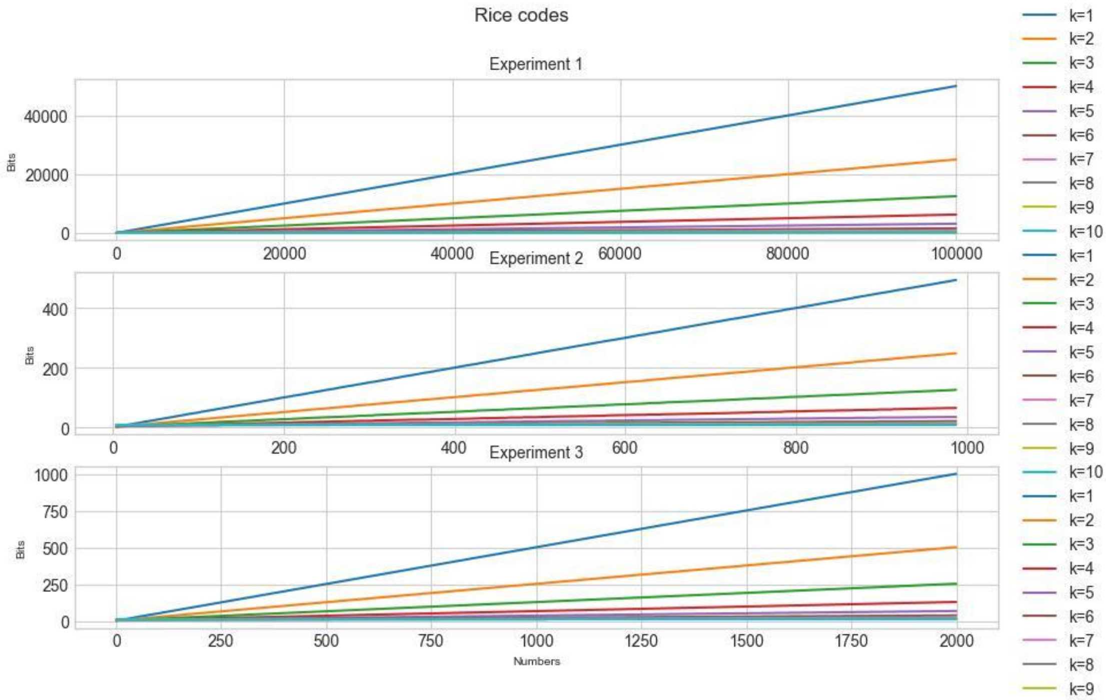

Figure 7: Plotting of bits used for value k chosen – Rice codes.

Figure 7 shows the three experiments in which there was applied a rice code with a specific value of *k*. As easy to see, in all of them, the number of bits used to encode each number grows linearly and takes over the 100 bits. As the integer grows, the number of bits used by the code grows, independently of the chosen value of *k*.

On the other hand, as the value of *k* grows, the number of bits used grows with a lower speed until it is possible to not differentiate the results from a value of *k* equals to 5.

Also, the statistics were studied for each experiment, that Figure 8 reports below. In the experiments 1 and 3, there is a common minimum equal to 2 bits used, while a value of 4 bits used in the experiment2.

Meanwhile, the maximum value obtained is 49997 bits for experiment 1, 494 bits for experiment 2, and 1001 bits for experiment 3.

About the median, there is a value equal to 785, 14, and 22 respectively for experiment 1, 2 and 3. The IQR has value 4854.5 for the experiment 1, 42.0 for the experiment 2 and 92.0 for the experiment 3.

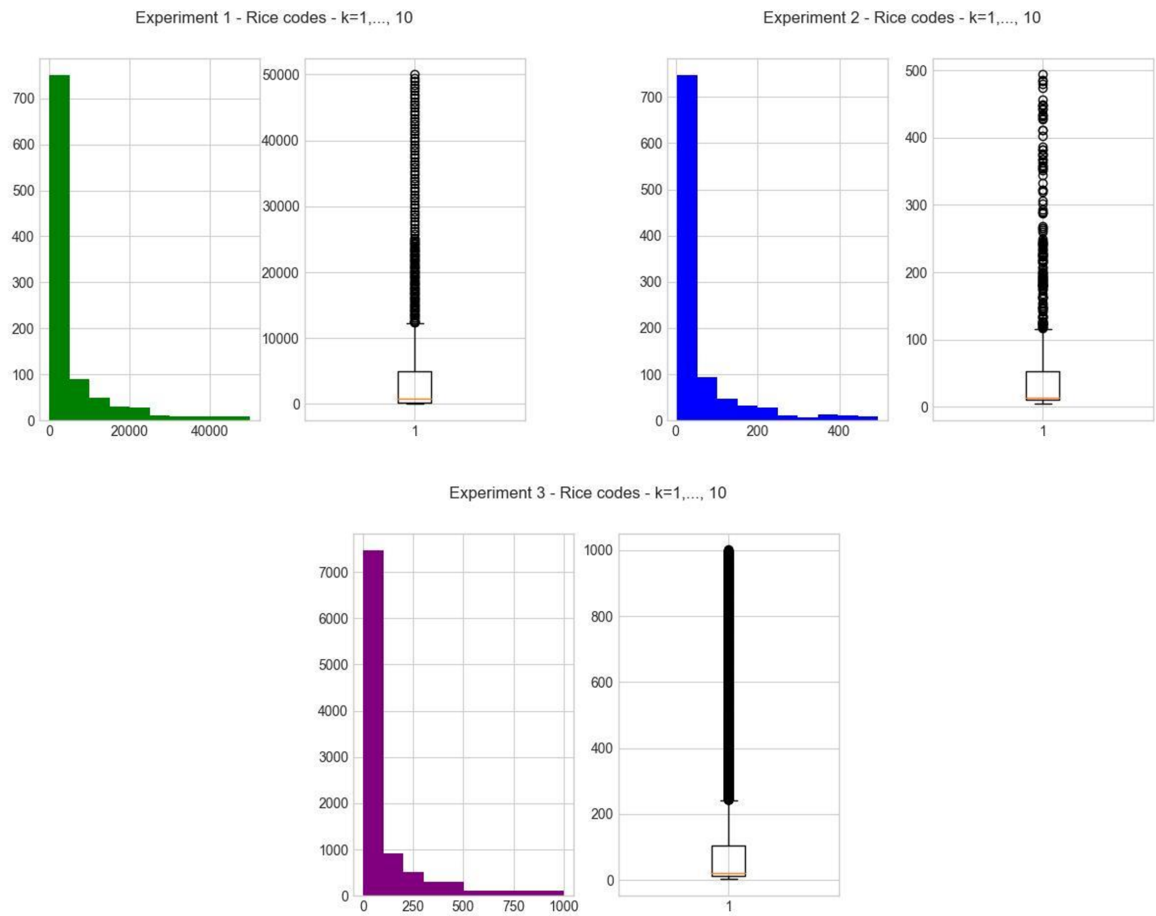

Figure 8: Rice codes – experiment 1 (green), experiment 2 (blue), experiment 3 (purple).

It is possible to see how the distributions are asymmetric and dense on the left part, involving a larger number of bits for the first about 200 integers.

CONCLUSIONS

The overall analysis is summarized in the following Figure 9.

Statistics Experiment1:

	Minimum	Maximum	Median	IQR
Gamma code	1	33	31.0	4.00
Delta code	1	25	24.0	4.00
Fibonacci code	2	25	24.0	2.25
Levenshtein code	2	29	24.0	6.00

Statistics Experiment2:

	Minimum	Maximum	Median	IQR
Gamma code	1	19	17.0	2.50
Delta code	1	16	15.0	1.25
Fibonacci code	2	16	14.0	2.00
Levenshtein code	2	18	17.0	1.50

Statistics Experiment3:

	Minimum	Maximum	Median	IQR
Gamma code	1	21	19.0	4.0
Delta code	1	17	16.0	2.0
Fibonacci code	2	17	16.0	2.0
Levenshtein code	2	19	18.0	2.0

Statistics Rice codes:

	Minimum	Maximum	Median	IQR
Experiment 1	2	49997	785.0	4854.5
Experiment 2	4	494	14.0	42.0
Experiment 3	2	1001	22.0	92.0

Figure 9: Table containing all the statistics for each experiment and Rice codes.

Based on these results, in order to use a smaller number of bits, it is recommended to use Delta and Fibonacci codes, which performances has showed the smallest values of the maximum obtained.