ESILV
ENGINEERING SCHOOL
DE VINCI PARIS

Blocks classification

Data analysis - Python project

Carole DUMOULIN
Mathieu GALIDIE

Problem

The problem consists of classifying all the blocks of the page layout of a document that has been detected by a segmentation process. This is an essential step in document analysis in order to separate text from graphic areas.

Indeed, the five classes are :

1. text
2. horizontal line
3. picture
4. vertical line
5. graphic

Dataset description

The 5473 examples comes from 54 distinct documents. Each observation concerns one block. All attributes are numeric. Data are in a format readable by C4.5.

There is no missing value.

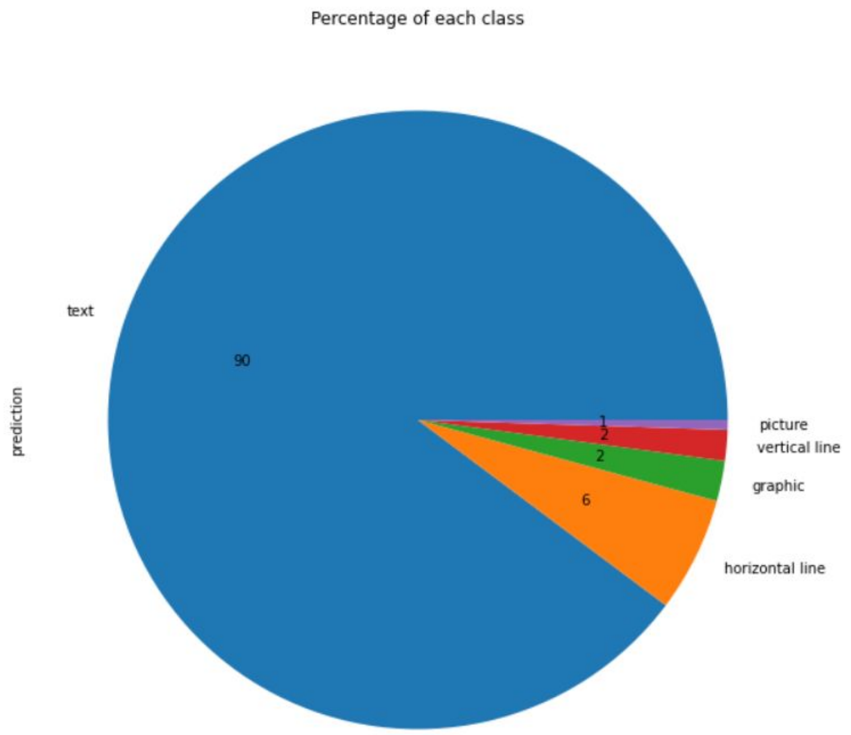
Each class is represented by the "prediction" attribute we have created.

Dataset attributes

Attribute	Type	Description	Domain
height	integer	Height of the block	[1 ; 804]
lenght	integer	Length of the block	[1 ; 553]
area	integer	Area of the block (height * lenght)	[7 ; 143993]
eccen	continuous	Eccentricity of the block (lenght / height)	[0,0070 ; 537,0]
p_black	continuous	Percentage of black pixels within the block (blackpix / area)	[0,052 ; 1,0]
p_and	continuous	Percentage of black pixels after the application of the RLSA (blackand / area)	[0,062 ; 1,0]
mean_tr	continuous	Mean number of white-black transitions (blackpix / wb_trans)	[1,0 ; 4955,0]
blackpix	integer	Total number of black pixels in the original bitmap of the block	[1 ; 33017]
blackand	integer	Total number of black pixels in the bitmap of the block after the RLSA	[7 ; 46133]
wb_trans	integer	Number of white-black transitions in the original bitmap of the block	[1 ; 3212]
prediction	integer	Block classes	{1 , 2, 3, 4, 5}

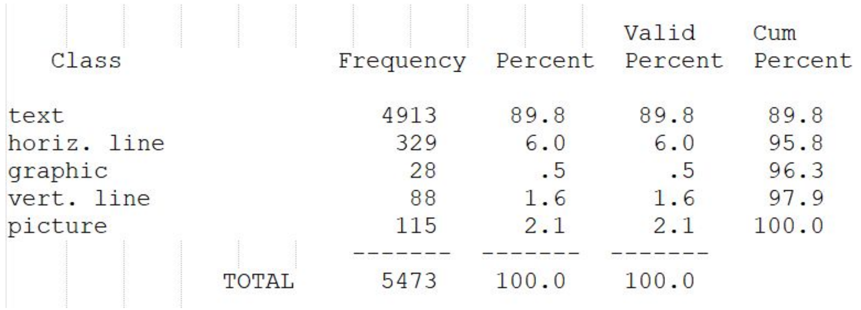
Data visualization

Data visualization



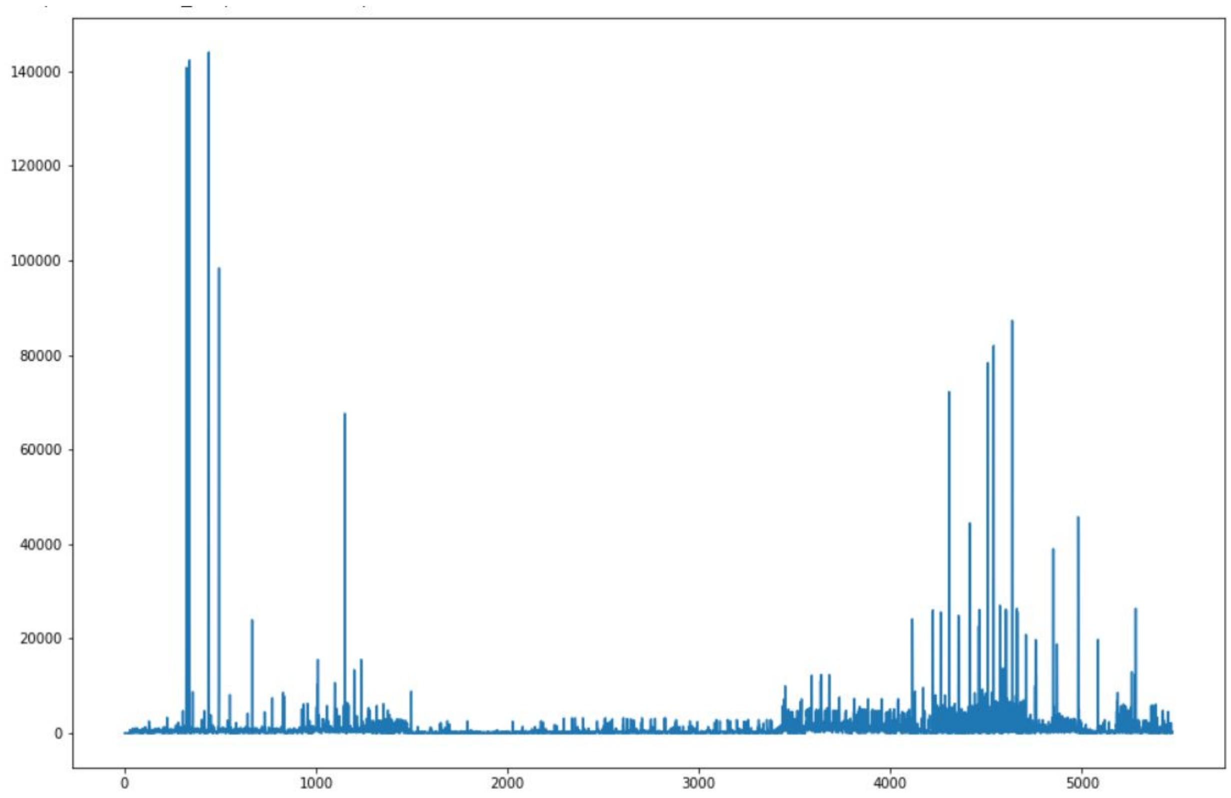
90% of the blocks are texts.

This is in harmony with the distribution of data provided in the description of the database.



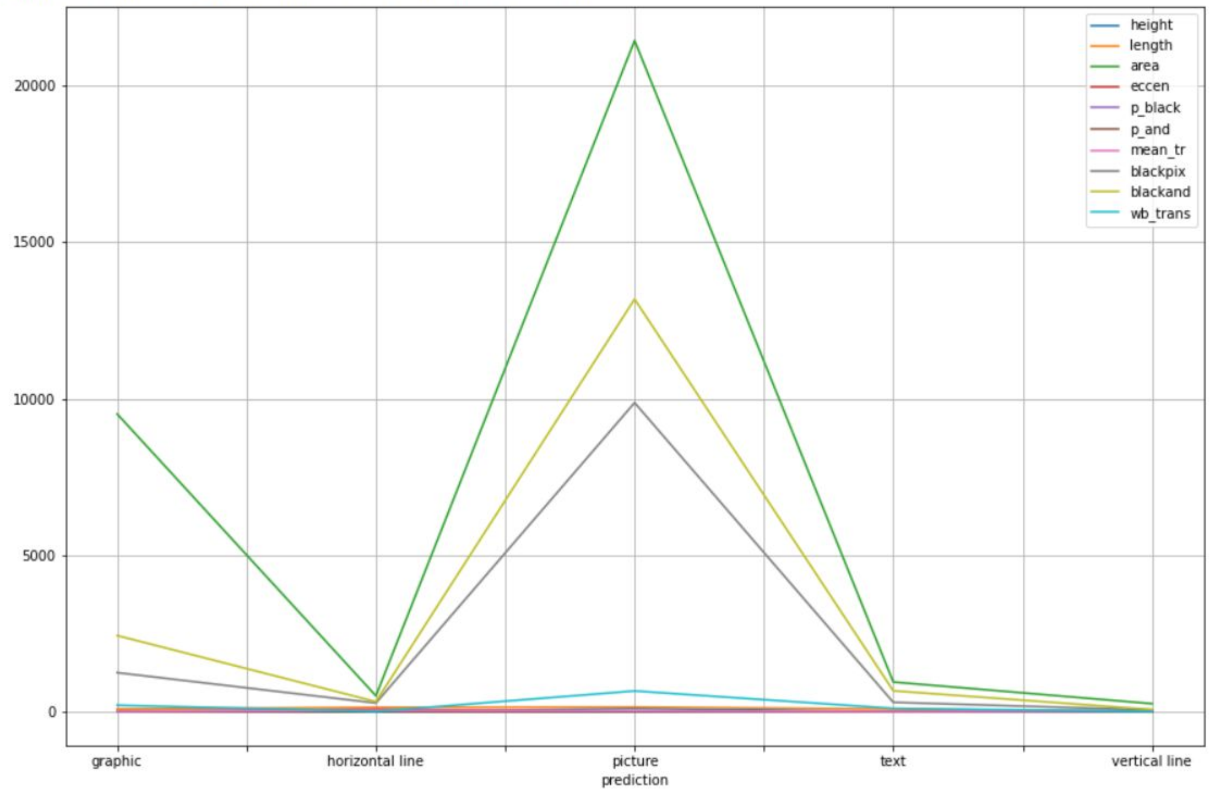
Class	Frequency	Percent	Valid Percent	Cum Percent
text	4913	89.8	89.8	89.8
horiz. line	329	6.0	6.0	95.8
graphic	28	.5	.5	96.3
vert. line	88	1.6	1.6	97.9
picture	115	2.1	2.1	100.0
TOTAL	5473	100.0	100.0	

Data visualization



99% of the blocks have an area content lower than 10 000.

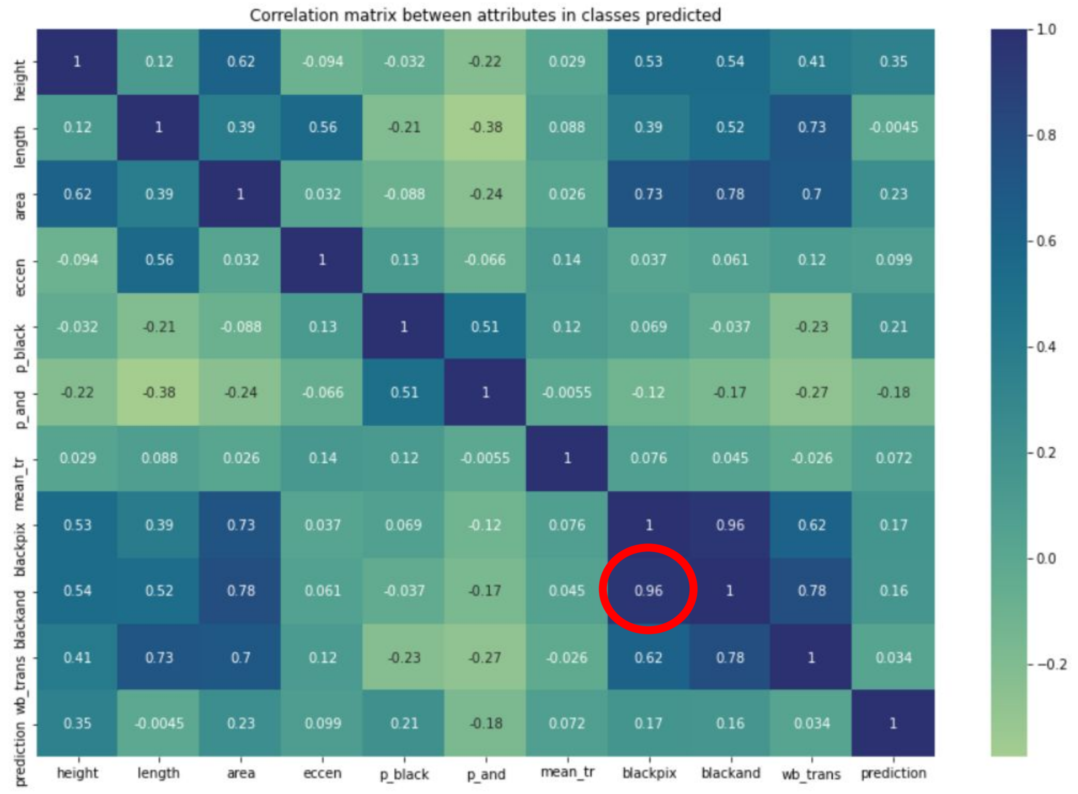
Data visualization



Pictures and graphics blocks take the most space (L*H area). (green curve)

Pictures have the higher total number of black pixels in the original bitmap of the block. There is a correlation between the number of pixels and the place taken by the block.

Data visualization



	height	length	area	eccen	p_black	p_and	mean_tr	blackpix	blackand	wb_trans	prediction
height	1	0.12	0.62	-0.094	-0.032	-0.22	0.029	0.53	0.54	0.41	0.35
length	0.12	1	0.39	0.56	-0.21	-0.38	0.088	0.39	0.52	0.73	-0.0045
area	0.62	0.39	1	0.032	-0.088	-0.24	0.026	0.73	0.78	0.7	0.23
eccen	-0.094	0.56	0.032	1	0.13	-0.066	0.14	0.037	0.061	0.12	0.099
p_black	-0.032	-0.21	-0.088	0.13	1	0.51	0.12	0.069	-0.037	-0.23	0.21
p_and	-0.22	-0.38	-0.24	-0.066	0.51	1	-0.0055	-0.12	-0.17	-0.27	-0.18
mean_tr	0.029	0.088	0.026	0.14	0.12	-0.0055	1	0.076	0.045	-0.026	0.072
blackpix	0.53	0.39	0.73	0.037	0.069	-0.12	0.076	1	0.96	0.62	0.17
blackand	0.54	0.52	0.78	0.061	-0.037	-0.17	0.045	0.96	1	0.78	0.16
wb_trans	0.41	0.73	0.7	0.12	-0.23	-0.27	-0.026	0.62	0.78	1	0.034
prediction	0.35	-0.0045	0.23	0.099	0.21	-0.18	0.072	0.17	0.16	0.034	1

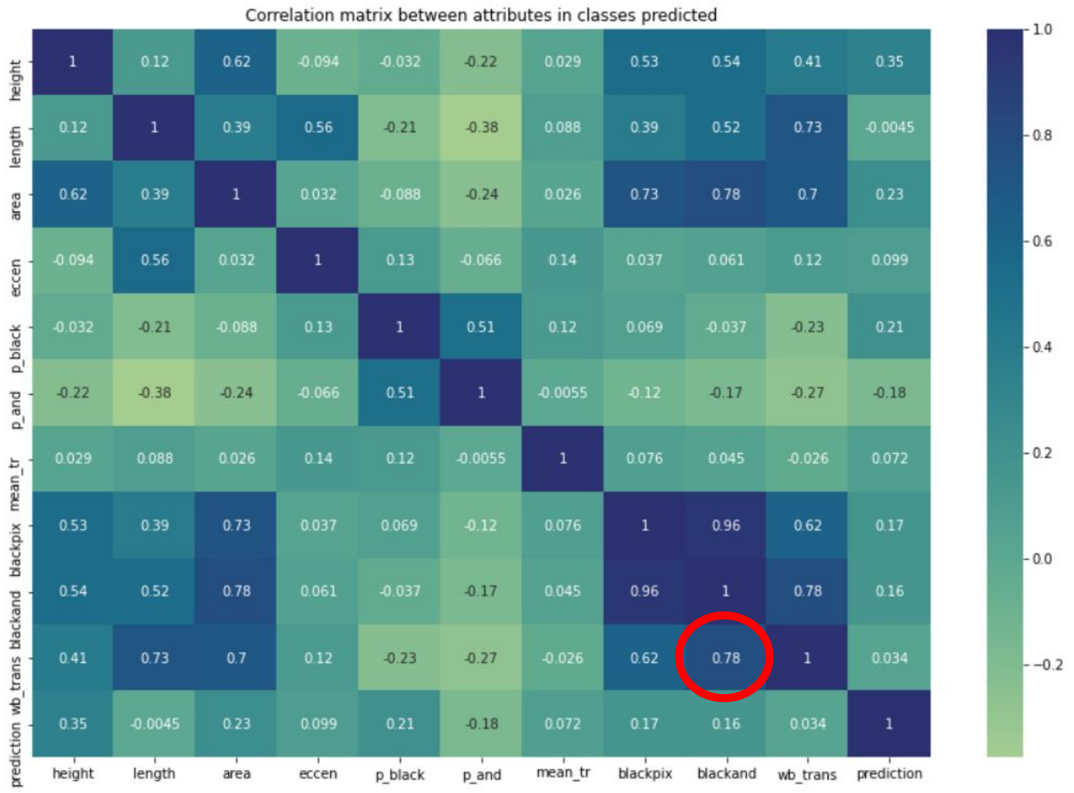
The highest correlation is between the blackpic and blackand attributes. These attributes represent the total number of black pixels in the original bitmap of the block and the total number of black pixels in the bitmap of the block after the RLSA.

This correlation is logical given the behaviour of the algorithm.

The basic RLSA is applied to a binary sequence in which white pixels are represented by 0's and black pixels by 1's. The algorithm transforms a binary sequence x into an output sequence y according to the following rules :

1. 0's in x are changed to 1's in y if the number of adjacent 0's is less than or equal to a predefined limit C.
2. 1's in x are unchanged in y .

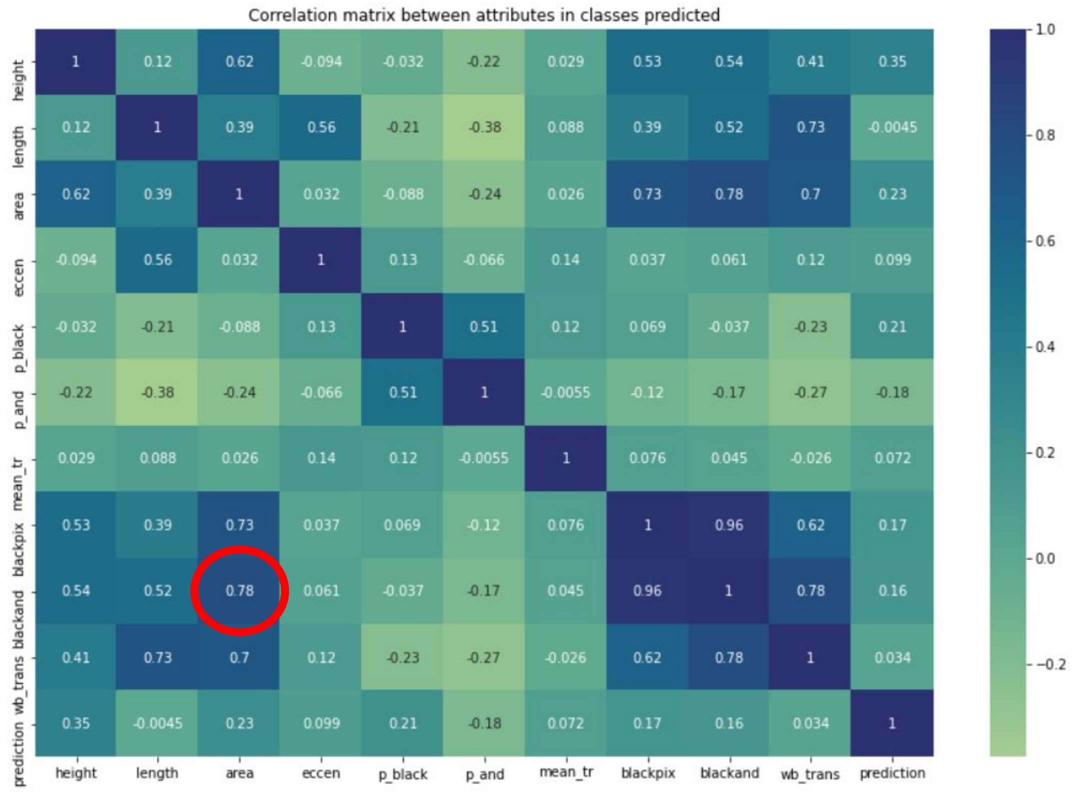
Data visualization



	height	length	area	eccen	p_black	p_and	mean_tr	blackpix	blackand	wb_trans	prediction
height	1	0.12	0.62	-0.094	-0.032	-0.22	0.029	0.53	0.54	0.41	0.35
length	0.12	1	0.39	0.56	-0.21	-0.38	0.088	0.39	0.52	0.73	-0.0045
area	0.62	0.39	1	0.032	-0.088	-0.24	0.026	0.73	0.78	0.7	0.23
eccen	-0.094	0.56	0.032	1	0.13	-0.066	0.14	0.037	0.061	0.12	0.099
p_black	-0.032	-0.21	-0.088	0.13	1	0.51	0.12	0.069	-0.037	-0.23	0.21
p_and	-0.22	-0.38	-0.24	-0.066	0.51	1	-0.0055	-0.12	-0.17	-0.27	-0.18
mean_tr	0.029	0.088	0.026	0.14	0.12	-0.0055	1	0.076	0.045	-0.026	0.072
blackpix	0.53	0.39	0.73	0.037	0.069	-0.12	0.076	1	0.96	0.62	0.17
blackand	0.54	0.52	0.78	0.061	-0.037	-0.17	0.045	0.96	1	0.78	0.16
wb_trans	0.41	0.73	0.7	0.12	-0.23	-0.27	-0.026	0.62	0.78	1	0.034
prediction	0.35	-0.0045	0.23	0.099	0.21	-0.18	0.072	0.17	0.16	0.034	1

The second highest correlation is between wb_trans and blackand attributes. These attributes represent the number of white-black transitions in the original bitmap of the block and the total number of black pixels in the bitmap of the block after the RLSA. This correlation is also logical given the behaviour of the algorithm described below.

Data visualization



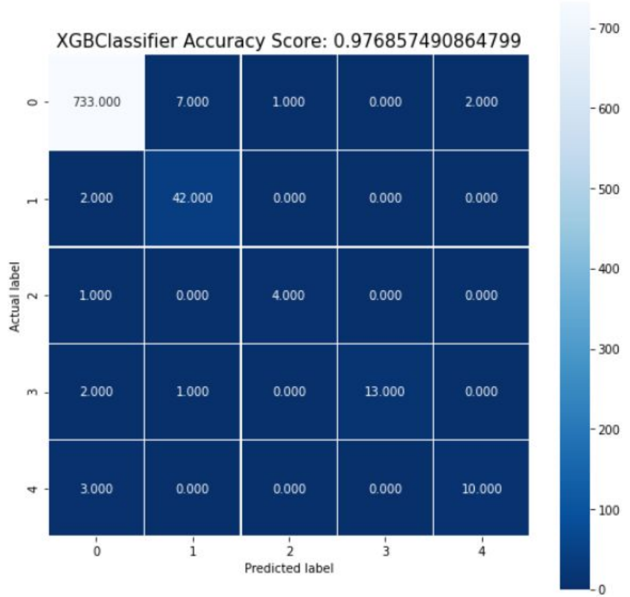
	height	length	area	eccen	p_black	p_and	mean_tr	blackpix	blackand	wb_trans	prediction
height	1	0.12	0.62	-0.094	-0.032	-0.22	0.029	0.53	0.54	0.41	0.35
length	0.12	1	0.39	0.56	-0.21	-0.38	0.088	0.39	0.52	0.73	-0.0045
area	0.62	0.39	1	0.032	-0.088	-0.24	0.026	0.73	0.78	0.7	0.23
eccen	-0.094	0.56	0.032	1	0.13	-0.066	0.14	0.037	0.061	0.12	0.099
p_black	-0.032	-0.21	-0.088	0.13	1	0.51	0.12	0.069	-0.037	-0.23	0.21
p_and	-0.22	-0.38	-0.24	-0.066	0.51	1	-0.0055	-0.12	-0.17	-0.27	-0.18
mean_tr	0.029	0.088	0.026	0.14	0.12	-0.0055	1	0.076	0.045	-0.026	0.072
blackpix	0.53	0.39	0.73	0.037	0.069	-0.12	0.076	1	0.96	0.62	0.17
blackand	0.54	0.52	0.78	0.061	-0.037	-0.17	0.045	0.96	1	0.78	0.16
wb_trans	0.41	0.73	0.7	0.12	-0.23	-0.27	-0.026	0.62	0.78	1	0.034
prediction	0.35	-0.0045	0.23	0.099	0.21	-0.18	0.072	0.17	0.16	0.034	1

The third correlation is between blackand et area attributes. These attributes represent the total number of black pixels in the bitmap of the block after the RLSA and the area of the block (height * length). This is a positive correlation. The more black pixels there are after the algorithm RLSA, the larger the area taken is.

Data modeling

BaseLine

We took 85% of our data for train and 15% for test.
We created a baseline so we can compare if our changes to the data/model impact the accuray



Actual \ Predicted	0	1	2	3	4
0	733.000	7.000	1.000	0.000	2.000
1	2.000	42.000	0.000	0.000	0.000
2	1.000	0.000	4.000	0.000	0.000
3	2.000	1.000	0.000	13.000	0.000
4	3.000	0.000	0.000	0.000	10.000

As you can see our BaseLine is 0.97 with XGBClassifier.

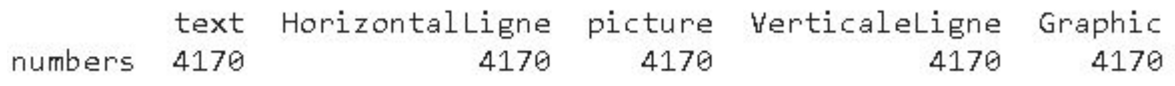
Preprocessing

Text class is 90% of all the data, and some of the class like graphic is 0.5 (28 over 5473). we need to have more balanced data in order to improve our model.

to deal with this imbalanced we have to have the same amount of data for each class. We can either duplicate data to have the same amount as the most one, or cut data to have the same amount as the lower one.

In this problem, the amount of data of our smallest one is so low that our model would be underFit (we only have 28 data for our lowest one)

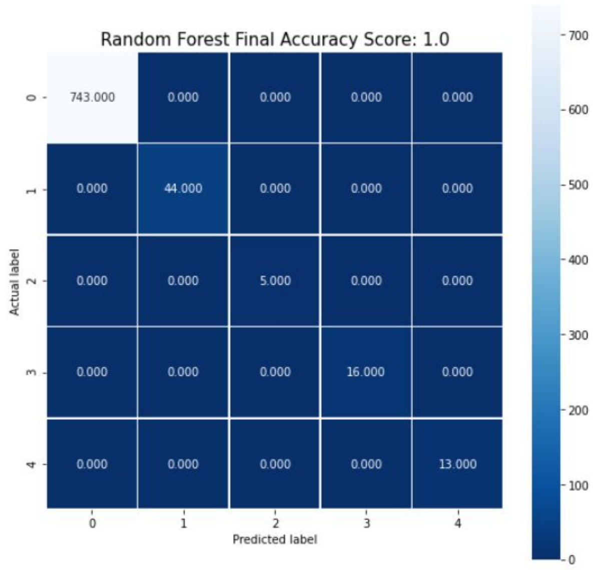
We chose to duplicate data until we have the same amount as our biggest class (text)



```
         text  HorizontalLigne  picture  VerticaleLigne  Graphic
numbers  4170             4170     4170            4170     4170
```

HyperParameters

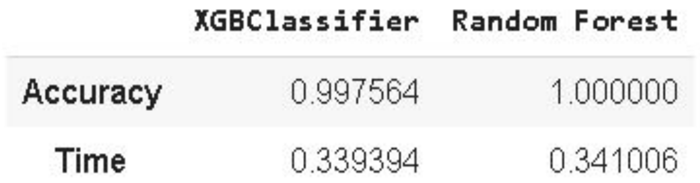
Our DataSet was so small that hyperParameters will not improve much the accuracy of our model, only the max_depth and n_estimators will changed something significantly.



Changing HyperParameters and duplicate Data has improved our accuracy from 0.9744 to 1.0000.

Choosing Production Model

Both model have almost the same accuracy, and the almost the same response time for 5000 request



	XGBClassifier	Random Forest
Accuracy	0.997564	1.000000
Time	0.339394	0.341006

Since Random Forest as 100% of accuracy and almost the same time for resquest as XGBClassifier, we will use Random forest for ou production Model

Conclusion

Conclusion

Our dataset was largely composed of texts. So, it was small and imbalanced. We managed to fixed imbalanced data by duplicate them.

We managed to score an accuracy of 100% which is very rare for a model.