PROJET DE PYTHON FOR DATA ANALYSIS

GAZEAU RÉMI

PRÉSENTATION DU DATASET

height:	integer.	Height of the block.
lenght:	integer.	Length of the block.
area:	integer.	Area of the block (height * lenght);
eccen:	continuous.	Eccentricity of the block (lenght / height);
p_black:	continuous.	Percentage of black pixels within the block (blackpix / area);
p_and:	continuous.	Percentage of black pixels after the application of the Run Length Smoothing Algorithm (RLSA) (blackand / area)
mean_tr:	continuous.	Mean number of white-black transitions (blackpix / wb_trans);
blackpix:	integer.	Total number of black pixels in the original bitmap of the block.
blackand:	integer.	Total number of black pixels in the bitmap of the block after the RLSA.
wb_trans:	integer.	Number of white-black transitions in the original bitmap of the block.

To Predict :

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	

OBJECTIF

- Réussir à déterminer à quoi correspond un élément dans une page internet

- Cette tâche est importante dans le cadre de l ’analyse automatique de documents

ÉTAPES

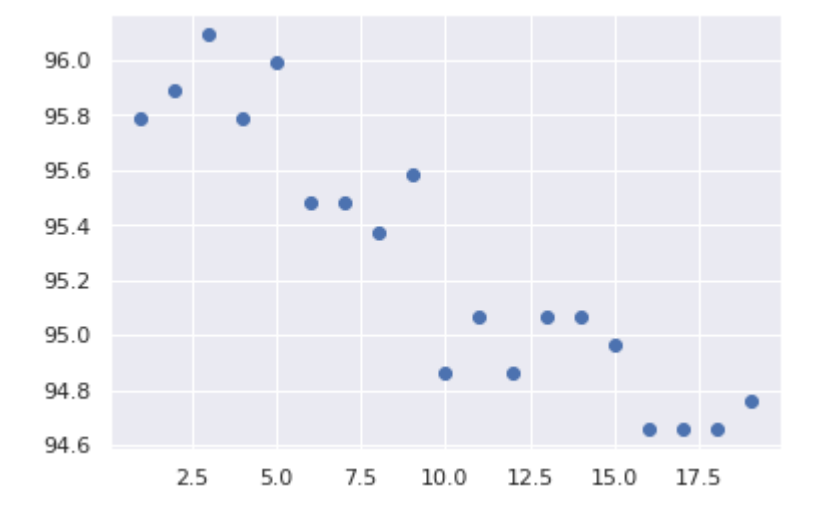
- Tester plusieurs modèles et plusieurs hyperparamètres
- Regarder si certaines variables semblent être délétères pour la performance du modèle
- Faire de nouveau un test avec le jeu de données nettoyé
- Tenter de voir si les performances sont meilleures après une ACP

MODÈLES TESTÉS

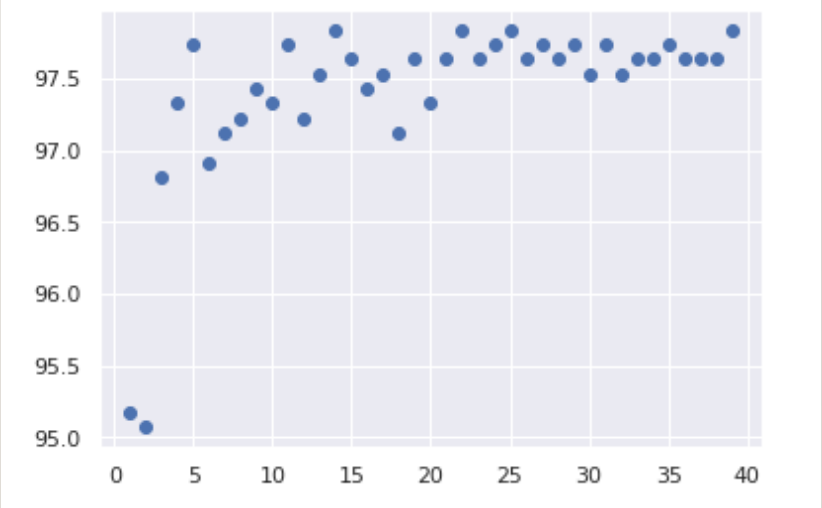
- KNN
- Nearets Centroid
- Decision Tree
- SVM
- Neural Networks
- Random Forest

MEILLEURS RÉSULTATS

KNN

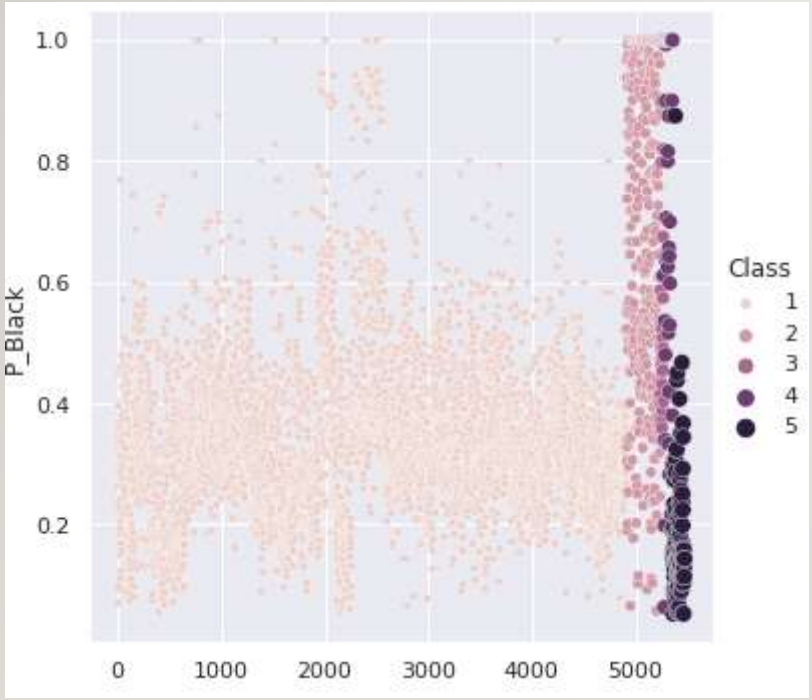


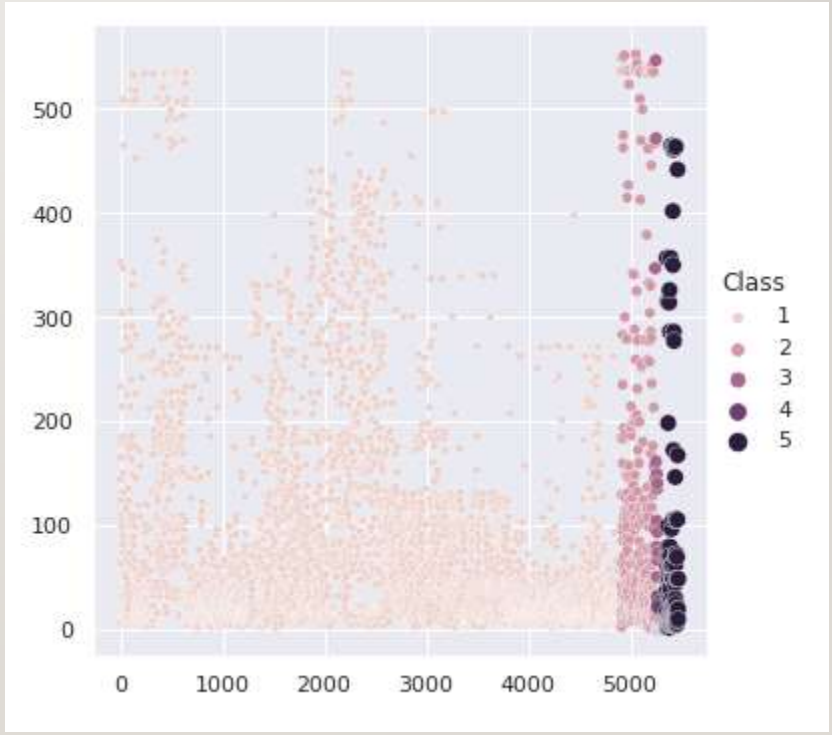
Random Forest



MODIFICATION DU DATASET

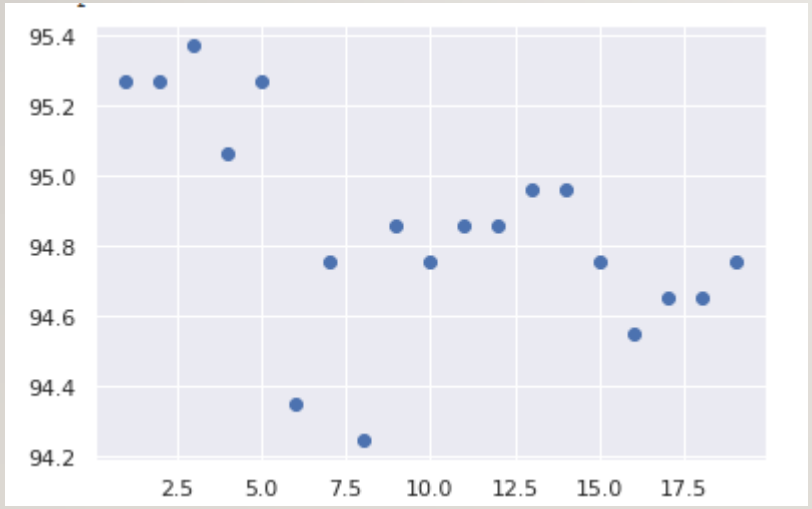
- Variables qui semblent ne pas apporter d'informations (Length et P_Black) => Suppression



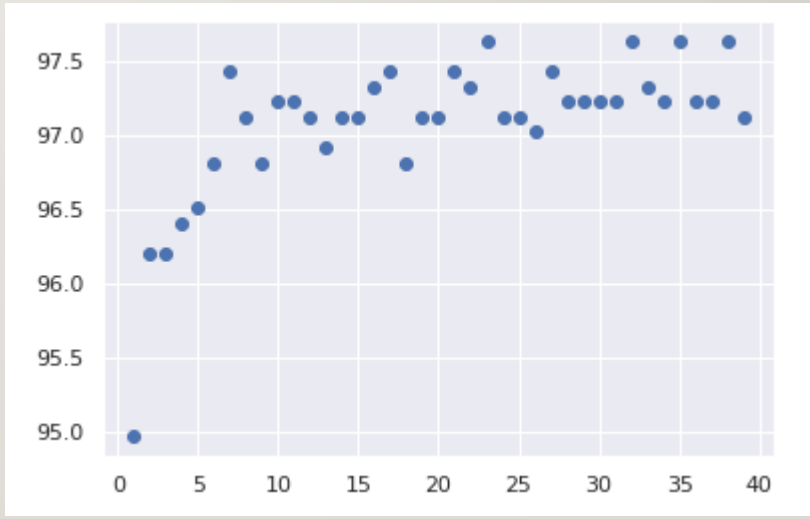


NOUVEAUX RÉSULTATS

KNN

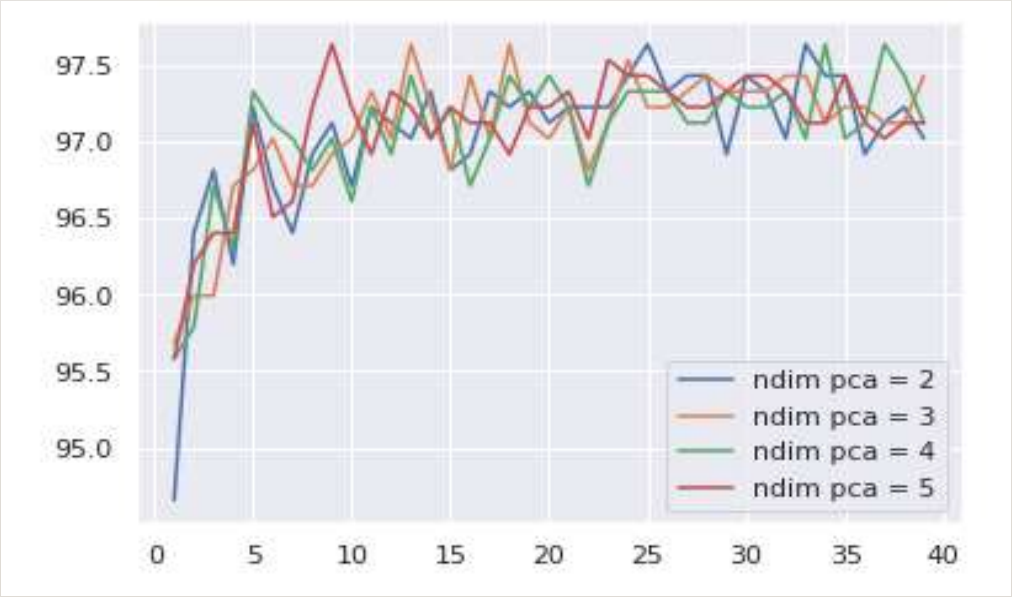


Random Forest



REDUCTION DE DIMENSION

- Résultats de la methode Random Forest suite à un de réduction de dimension



CHOIX DU MODÈLE

- Méthode de la Random Forest
- Nombre d'arbres : 25%
- Performance : environ 97,5% de réussite

API

- Données à mettre dans un fichier CSV
- Utilisation de la librairie Flask
- Modèle exporté au format Pickle