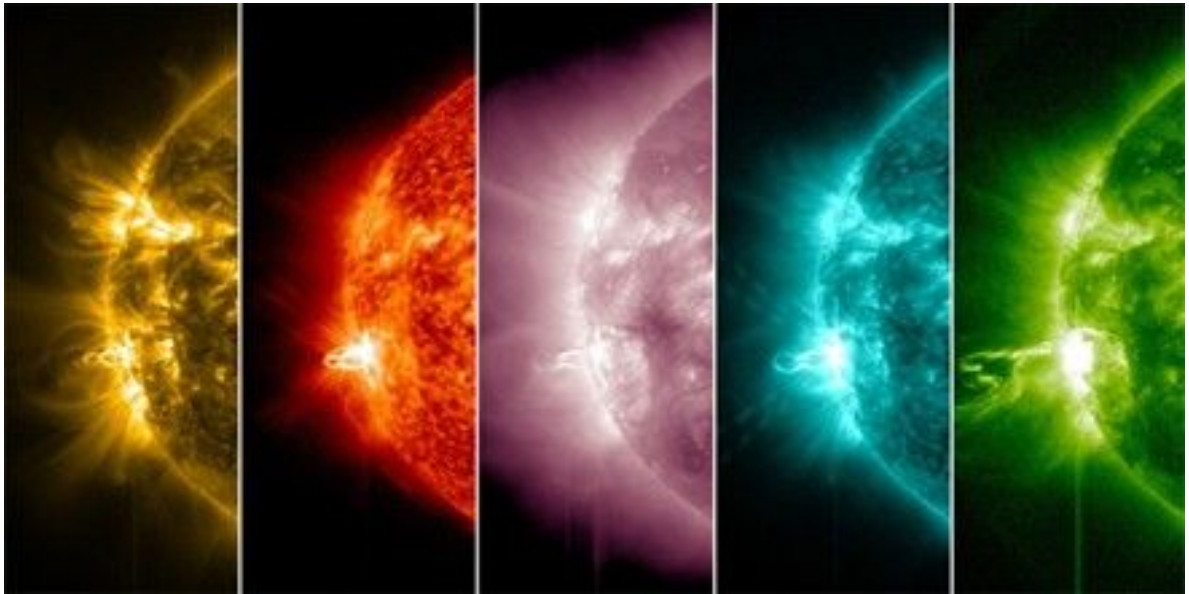
SOLAR FLARE DATASET

From UCI Machine learning: https://archive.ics.uci.edu/ml/datasets/Solar+Flare

Final Project Ironhack Data Analytics 2020



Michael Wenzelhumer

Objective

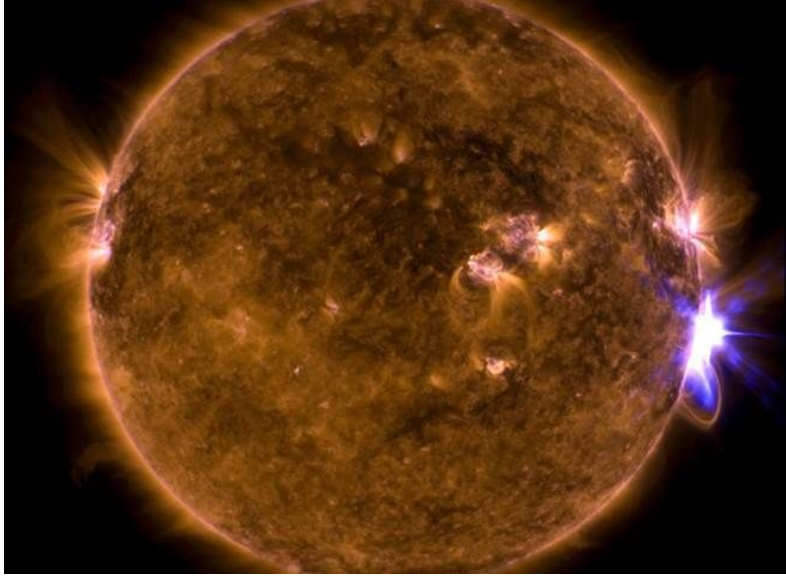
Three class types

C-class (common flares)

M-class (moderate flares)

X-class (severe flares)

Method



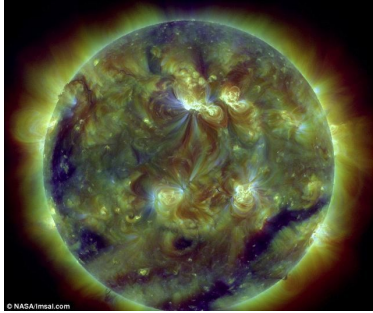
10 attributes to measure

3 categorical columns----> created dummies

Removed 1 column(Area of largest spot) due to redundancy of data

1. Code for class (modified Zurich class) (A,B,C,D,E,F,H)
2. Code for largest spot size (X,R,S,A,H,K)
3. Code for spot distribution (X,O,I,C)
4. Activity (1 = reduced, 2 = unchanged)
5. Evolution (1 = decay, 2 = no growth, 3 = growth)
6. Previous 24 hour flare activity code (1 = nothing as big as an M1, 2 = one M1, 3 = more activity than one M1)
7. Historically-complex (1 = Yes, 2 = No)
8. Did region become historically complex on this pass across the sun's disk (1 = yes, 2 = no)
9. Area (1 = small, 2 = large)
10. Area of the largest spot (1 = <=5, 2 = >5) removed

Method



Dataframe with dummies

	Class	Spot Size	Spot Distribution	Activity	Evolution	24hr activity	Historically-complex	Become complex on this pass	Area	C-class	...	Spot Size_A	Spot Size_H	Spot Size_K	Spot Size_R	Spot Size_S	Spot Size_X	Distrit
0	H	A	X	1	3	1	1	1	1	0	...	1	0	0	0	0	0	
1	D	R	O	1	3	1	1	2	1	0	...	0	0	0	1	0	0	
2	C	S	O	1	3	1	1	2	1	0	...	0	0	0	0	1	0	
3	H	R	X	1	2	1	1	1	1	0	...	0	0	0	1	0	0	
4	H	S	X	1	1	1	1	2	1	0	...	0	0	0	0	1	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
1061	H	S	X	1	2	1	1	1	1	0	...	0	0	0	0	1	0	
1062	H	S	X	2	2	1	1	2	1	0	...	0	0	0	0	1	0	
1063	C	S	O	1	2	1	2	2	1	0	...	0	0	0	0	1	0	
1064	H	R	X	1	2	1	1	2	1	0	...	0	0	0	1	0	0	
1065	B	X	O	1	1	1	1	2	1	0	...	0	0	0	0	0	1	

1066 rows × 28 columns

Some Corrs..

-*Spot size_x*

Class B

-*Class A*

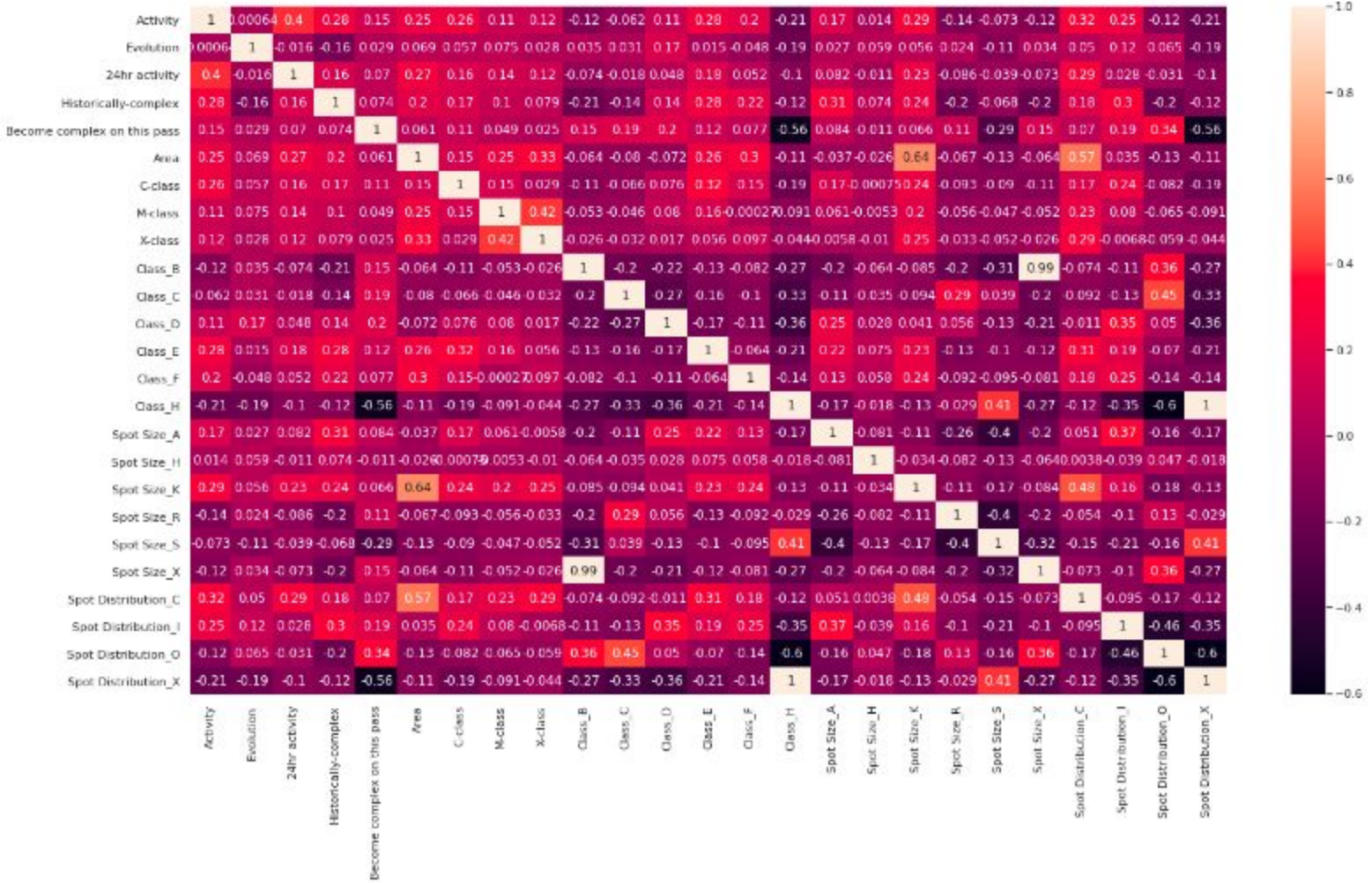
Spot dist_x

-*Class_H*

Spot_dist_x

-*Spot size_x*

Class_b



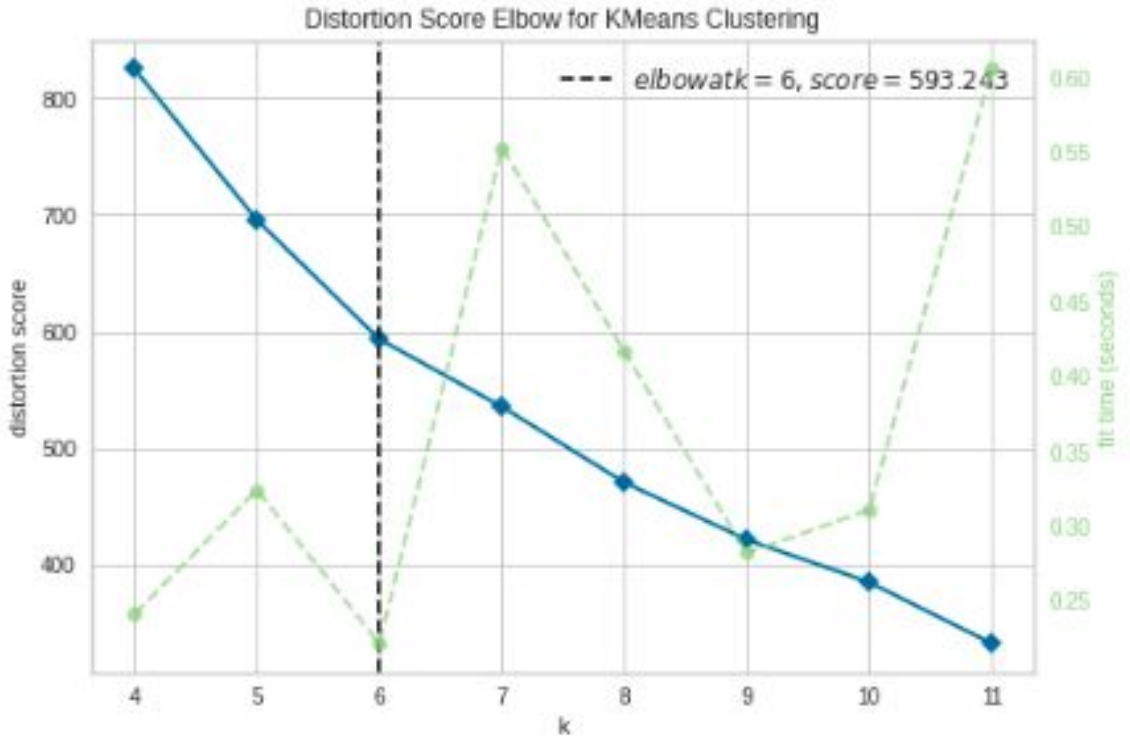
Kmeans

Confusion Matrix after clustering:

[[235 36]
[26 23]]

Accuracy: 0.80625

-Clusters increased to 6



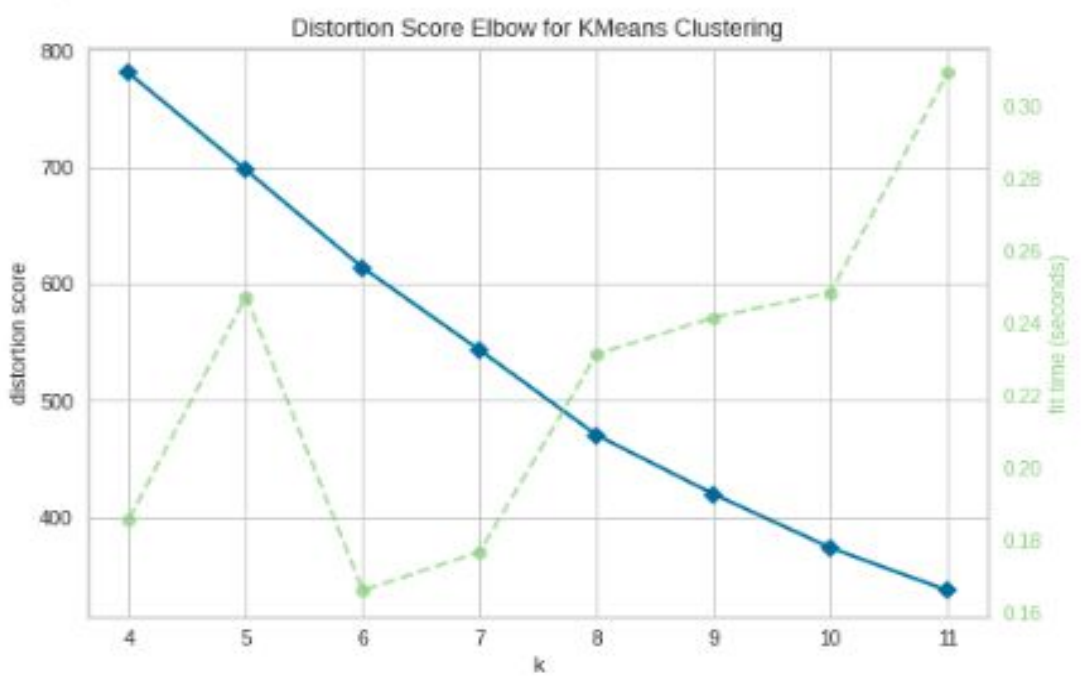
Kmeans

Improvement in confusion matrix

[[259 12]

 [41   8]]

Accuracy: 0.834375



Results

Train Test split

RandomForestClassifier=0.79375

DecisionTreeClassifier= 0.784375

GaussianNB= is 0.80625

Clustering

RandomForestClassifier=0.796875

DecisionTreeClassifier= 0.809375

GaussianNB= 0.75

PCA

RandomForestClassifier=0.784375

DecisionTreeClassifier= 0.7875

GaussianNB= 0.7625

Method Pycaret

Application of PyCaret-

KNN-0.8068

RandomForest-0.804

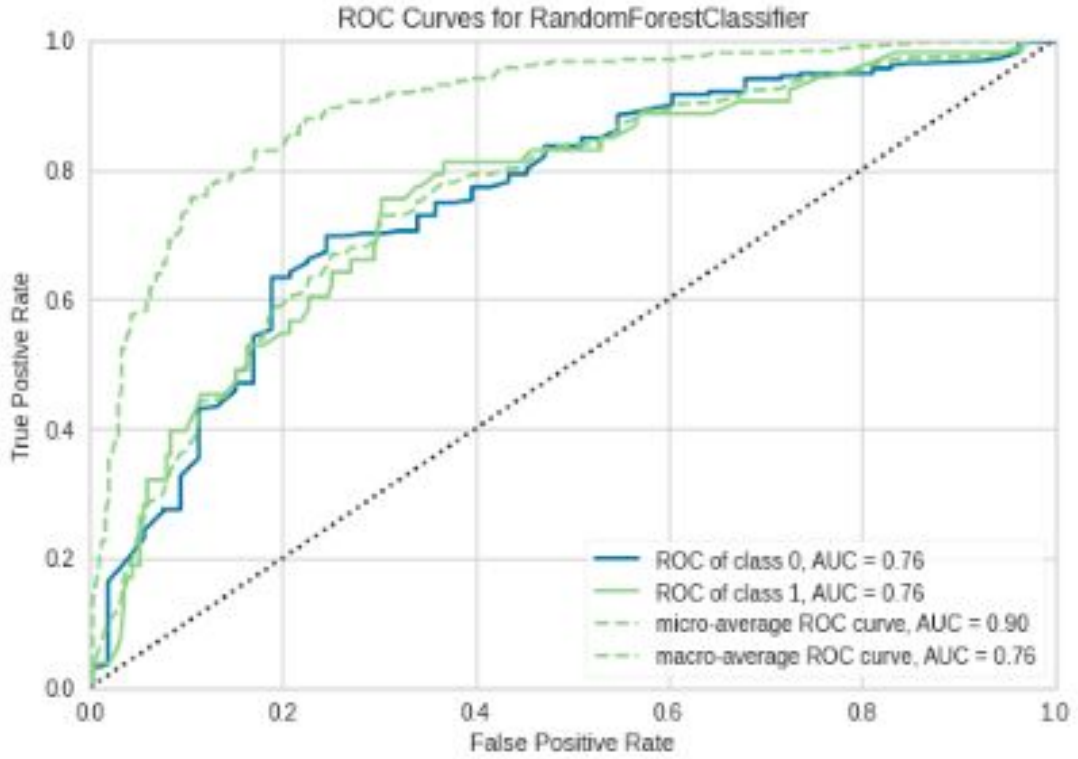
NaiveBayes-0.802

DecisionTree-0.7983

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Logistic Regression	0.826500	0.738800	0.173100	0.552100	0.251100	0.181400
1	Ridge Classifier	0.825100	0.000000	0.148700	0.483300	0.218900	0.154500
2	Ada Boost Classifier	0.815300	0.721500	0.156400	0.477900	0.223500	0.144200
3	Gradient Boosting Classifier	0.808200	0.691500	0.156400	0.324500	0.201200	0.119200
4	Light Gradient Boosting Machine	0.808200	0.688700	0.172400	0.329400	0.220400	0.134200
5	K Neighbors Classifier	0.806800	0.663200	0.107100	0.261700	0.151100	0.075300
6	Random Forest Classifier	0.804000	0.671100	0.196800	0.378300	0.249700	0.152600
7	Extra Trees Classifier	0.804000	0.641900	0.164100	0.289500	0.207200	0.119300
8	CatBoost Classifier	0.804000	0.663900	0.162800	0.334000	0.211500	0.121000
9	Naive Bayes	0.802600	0.731700	0.330800	0.406000	0.344700	0.235400
10	Extreme Gradient Boosting	0.802600	0.711200	0.132100	0.315000	0.179800	0.092900
11	Linear Discriminant Analysis	0.801100	0.722400	0.222400	0.371600	0.272200	0.168500
12	Decision Tree Classifier	0.798300	0.582400	0.180800	0.347700	0.232800	0.131400
13	SVM - Linear Kernel	0.764200	0.000000	0.255800	0.248100	0.222200	0.120500
14	Quadratic Discriminant Analysis	0.428900	0.636400	0.703800	0.203500	0.295500	0.050300

Visualization

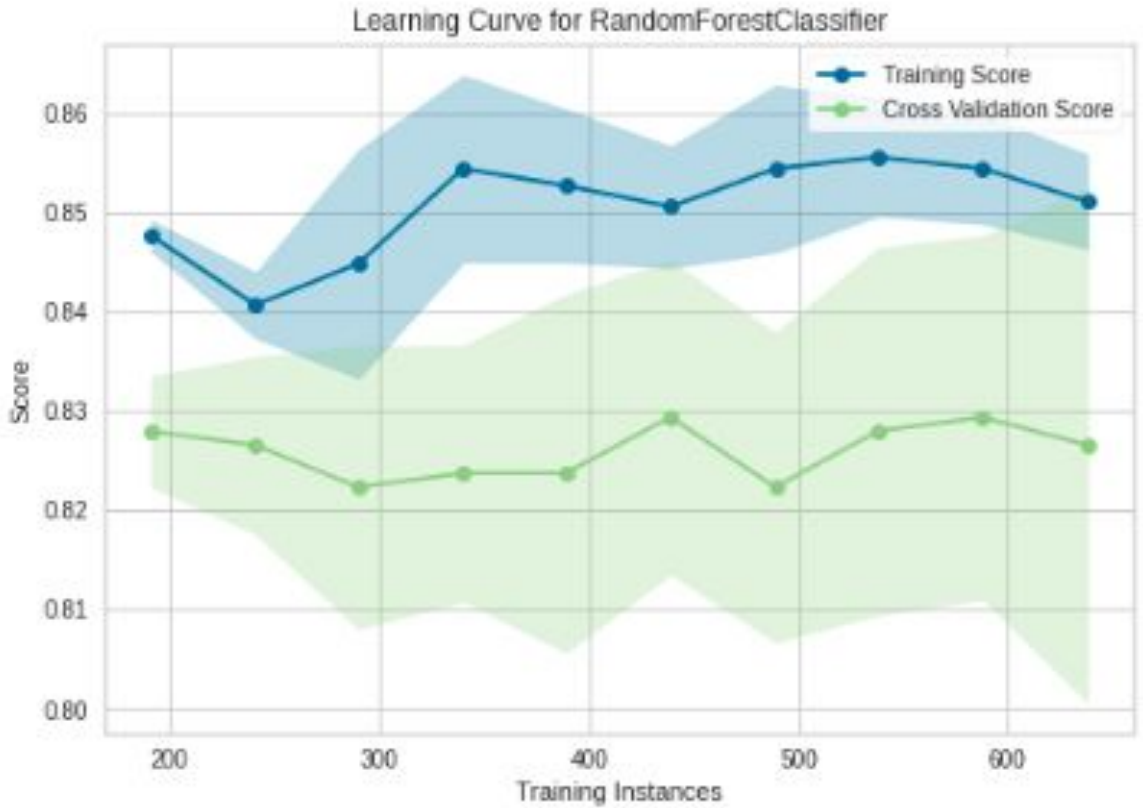
Tuned RF-



Visualization

-Learning curve

For RF



Conclusion

- Decision Tree classifier proved to be most accurate model after clustering
- Need to create more models for class M and X flares
- Creation of a more complete pipeline
- Closer comparison of Pycaret model

  With standard models.