Hi,

So I started off with parsing all the files from the first partition and extracting basic information from it and combined into a dataframe.

I first processed all the flare files , calculated basic descriptive stats on them and put them in a dataframe. I did the same for the non-flare samples and then concatenated the two dataframe.

**(GO TO NOTEBOOK AND POINT TO FINAL-DATA)**. BTW THIS IS MY ROUGH SPACE. THE DOCUMETED CODE IS ON MY REPOSITORY. HERE IS THE CODE FOR IT.

**(POINT TO CODE ON HELPER FUNCTION)**

Next, I computed basic stats on all the extracted features**.** such as the cardinality, null values, percentiles and a few more. The outlier count low is the number of features valus that lie below 1.5\*IQR range and the outlier count high is the number of feature values that lie above 1.5\* IQR. **(POINT TO summary\_table in the notebook)**

Here, are a few findings from the computed stats**. (POINT TO DASHBOARD SUMMARY PAGE).**

1. M and x class fares constitute less than 10% of the class distribution. I suppose, this pretty much represent the infrequent nature of solar flares or atleast the ones that affect the infracstructure on earth.
2. This graph is the total number of low and high outliers for feature. The read line represent the threshold for the maximum number of outlier allowed, which here is 10% of the number of samples in a feature. If the number of outliers exceed this threshold, its best to drop them. The threshold seleced can obviously vary.
3. The bar graph here represent the maximum and minimum values of each feature. The line represents the range. I have sorted this graph using range. Now, the features with large range and also a high number of outliers need to be transformed in some way. Typical transformations used for skewed data are log transform, square root transform , box-cox transform or even just clamping the feature value to a certain percentile. **(POINT TO TRANSFORMATIONS CLASS )**
4. Also, the features with few outlier and smaller range, are good candidates for normalization. **(POINT TO TRANSFORMATIONS CLASS)**
5. This graph describes the number of unique value for each of the extracted feature. Since, we are dealing with numeric data here, the cardinality must ideally be equal to the number of samples in the feature. Hence, features with low cardinality need to be assessed further prior to removing them to understand the reason for the same. Since, I don’t have a lot idea about what each feature does or how its calculated, I have left features even with high cardinality untouched.
6. **(POINT TO TRANSFORMATION DASHBOARD)** These are few of the feature prior to transformation and post transformation. As you see, the feature values are drastically scaled down. As explained before, the corresponding transformation is based on the number if outlier and range of every feature.
7. Next, I trained some baselines models and computed cross val scores on each of them. **(POING TO DATA\_CLEANING).**This is the code for it. I am yet to document this file using docstring, but I ll get it done by this week. These are just some helper function to produce plots , perform gris search and carry out k-fold cross validation. (**POINT TO SCORES DASHBOARD).**  Here are the results of the baseline models. I did not compute accuracy, it would anyways be misleasing cuz of the class imbalance. As you can see f1-scores and auc-scores both reduced with every subsequent fold. Again, these are baseline models without any kind of hyperparameter tuning or feature selection. Random forest had an auc score close to one, which is weird. I assume the reason for this is because the samples within a same partition are very similar.
8. Yeah, that’s about it really.