* Project Goals
  + What is your goal ?
  + What is the data about ? Where is it from ?
  + How will you approach the problem ?
* Exploratory Data Analysis
  + Is imputation needed i.e. are there NA’s ? If not still show your work. If yes clean the data.
  + Are there duplicates ? Should you clean them ? (DataFrame.duplicated() and DataFrame.drop\_duplicates() functions can help). We will discuss the effect of having duplicates
  + Discuss the input features. Which ones are categorical ? Which ones are not ?
  + Do you need to one hot encode the categoricals ? That will depend on the eventual algorithm you choose. For example any tree based algorithm (RandomForest, Gradient Boosters etc) can handle categoricals without requiring encoding. Other models will require encoding. Think about do you want to just pick one algorithm and stick with it ? Or do you want to do a few and compare ? No right or wrong way here.
  + Do correlation analysis between the input features and the label. You can use the corr() function on the dataframe. Draw correlation plots to show the work. Seaborn has some nice plots. What does the correlation tell you ? Are there input’s that have no correlation with the label ?
  + Is the data balanced i.e. do you have similar number of samples in all the label categories ? The Counter class or np.bincount() will come in handy here. If there is a lack of balance what does this mean ? We will talk about this in more details as well.
* Feature Selection
  + Are all features equally important ? The correlation work may give you a hint, but not the best way to select.
  + Look at feature selection in the sci-kit learn docs. We will talk about this in more details – but basically you can do chisquared tests (or f tests) to see which ones have a p-value less than 0.05 and pick those i.e. pick the significant ones.
* Modelling
  + Splitting the data. Think about what a right kind of split is. The dataset is pretty large – so chances are you do not need a large test percentage. Save your test data as a separate csv file – may come in handy later.
  + Are you going to compare multiple models ? Or just stick with one model ? In either case you should do a baseline training on the model(or models) you want to work with along with some hyper parameter searching and cross val. Your goal should be to see if you can then improve on the best validation score.
  + Choosing the right metric is very important. Think what might be the best scoring metric here. Is accuracy a good metric ? Or should you consider precision/recall ? What about both (via the F1 score) ? Or may be AUC score ? Will talk through this as well.
  + Check the returned scores to ensure you are not overfitting. If you are overfitting – try to regularize.
  + If you are comparing multiple models, you might want to pick the model with the best baseline score and stick with it – or try to improve all of them. Again no right or wrong answer – depends on how much time you have.
  + What kinds of things can you do to improve ? One approach could be to try and correct the imbalance. You can do it by specifying class weights – or you can do it via resampling – will talk about both.
  + If you have evaluated multiple models, yet another thing you can do is to try and create a voting classifier to combine the models and see if that improves the results.
  + Another approach could be to get more data. Can you use the notebook the Kaggle contributor used to grab data from other years ? (This data is 2015 I believe)
* Conclusion
  + What did you conclude ?
  + What other things can be done to improve quality that you did not get to do in the project ?