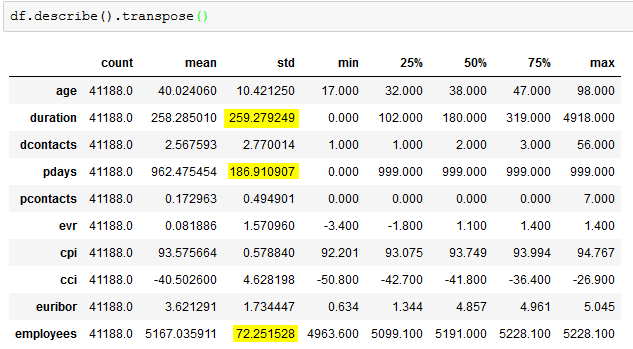
**1. Abstract**

Given some information of a marketing campaign, the goal of this project is to *predict whether or not they end up subscribing for a term deposit*.

**2. EDA**

***df.describe()***

Using describe() on the dataframe, for summary statistics of all the quantitative(numeric) variables.

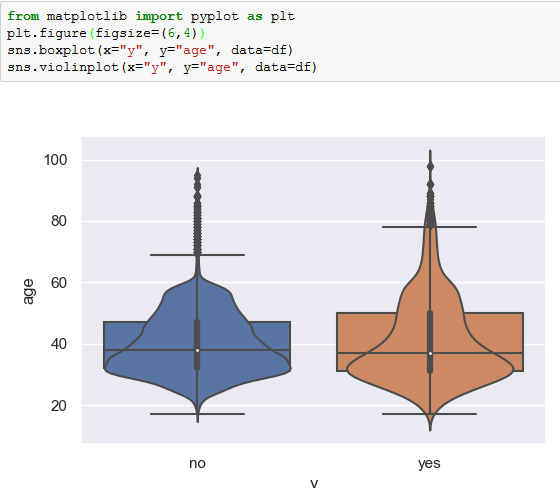


The standard deviations of 'duration', 'pdays' and 'employees' are very large compared to other variables. These variables should be investigated to understand the reasons for this variability.

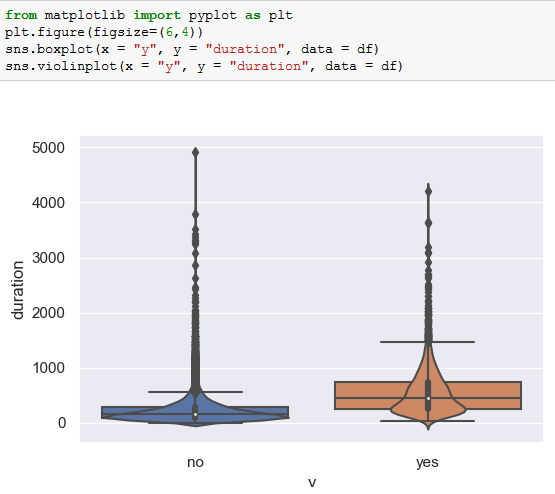
As part of Graphical EDA, I plot two graphs

* Histograms
* Violin plot

**2.1 EDA with Numeric Variables:**

**‘age:** The variance of age of the customers who have rejected the offer is lower compared to that of the customers who have accepted.

There are significant number of outliers for both classes. However, the outliers for 'no' are widespread. Binning the 'age' variable with respect to 'job' category might provide us better insights.

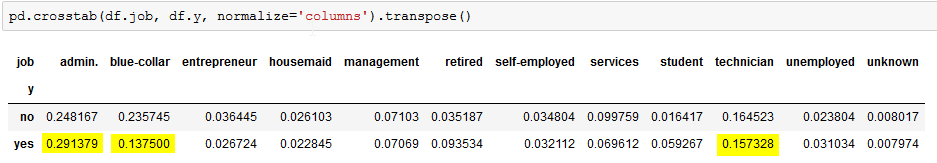
****

**‘duration’:** The variance of class 'no' of the response variable is less compared to that of 'yes' class. Outliers for 'no' are widespread than the outliers of 'yes'.

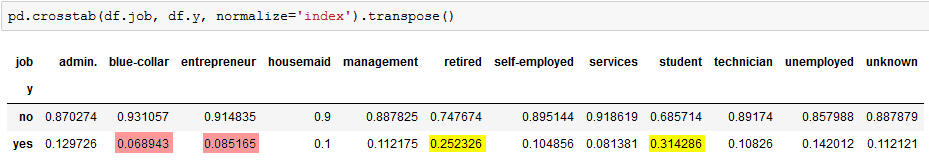
Since the data is widespread, it's a good idea to bin them and include upper bounds.

**2.2 EDA on Categorical Variables:**

**‘job’:**

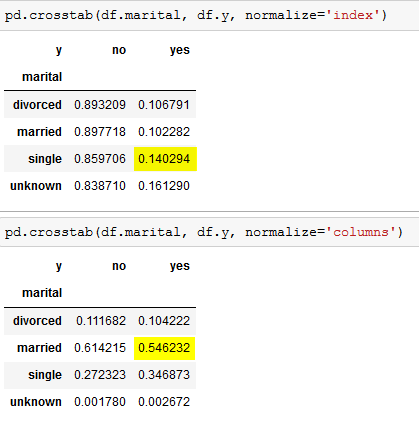
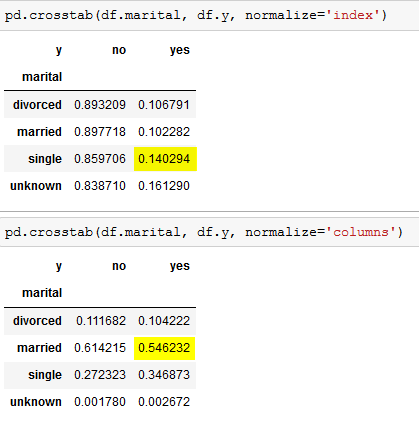


**Interpretation:** At category-level, Admin, Blue-Collar and Technicians contributed the highest percentage of positive response rate.

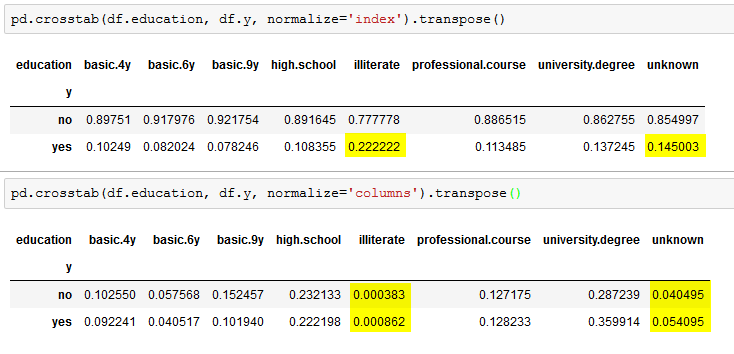


**Interpretation:** At class-level, ‘blue-collar’ and ‘entrepreneur’ (6% and 8%) had the lowest positive response rate while retired and students had high positive response rate (25% and 31%).   
Though ‘admin’ and ‘blue-collar’ jobs were contacted, the highest positive response rate, is among ‘retired’ and ‘student’, rather not ‘blue-collar’ and ‘entrepreneur’.

**‘marital’:**

   
**Interpretation:** At class-level, 'unknown' has the highest positive response rate. At category-level, with 55% 'married' contributed the highest percentage of positive response rate.

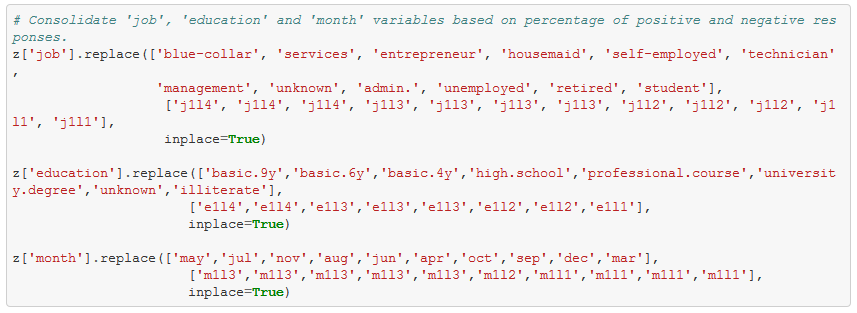
**‘education’:**



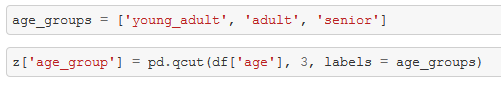
**Interpretation:** The distinction between the education descriptions is very minimal which makes it hard to combine similar classes in the category.

**3. Feature Engineering**

**3.1 Consolidate category classes:**



**3.2 Binning the age:**



**3.3. Categorize 'day' with 'weekday\_1', 'weekday\_2' and 'weekend' classes:**

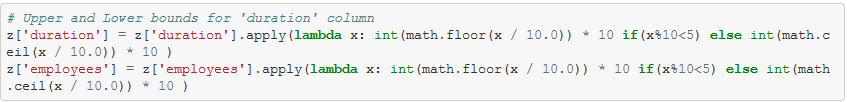


**3.4 Merging 'marital' and 'age' variable:** 

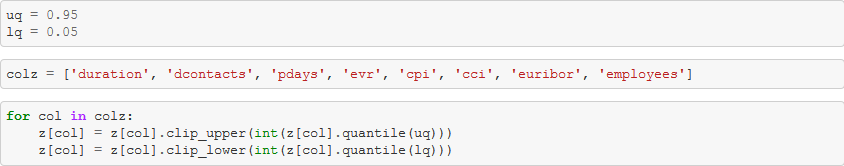
**3.5 Inclusion and Exclusion of 'duration' column:**

‘duration’ highly affects the output, I create two dataframes (one with 'duration' column, one without).

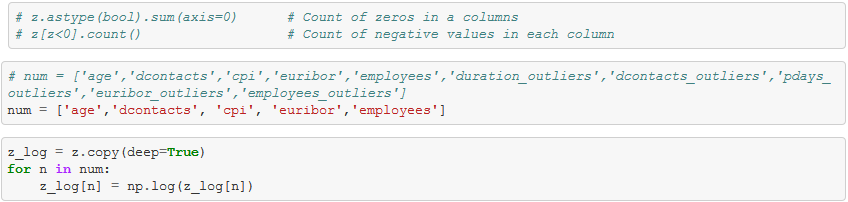
**3.6 Treating Outliers:  
3.6.1. Applying Upper and Lower bounds to 'duration' and 'employees' variable**



**3.6.2. Applying 90 percentiles and 5 percentiles for the lower and upper outliers**

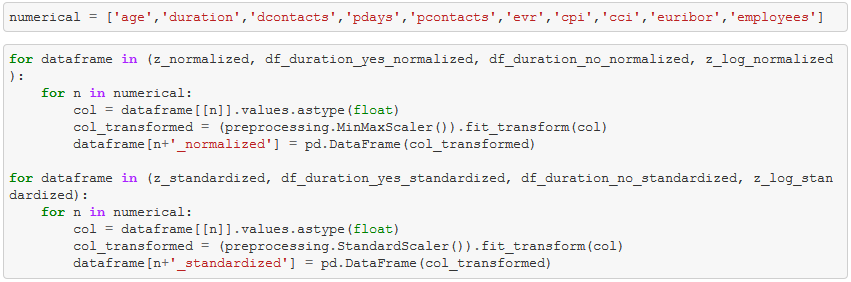


**3.6.3. Apply Logarithmic transformations to invalid outliers**Creating a new dataframe to apply logarithm transformations. From all the numerical columns, logarithmic transformations is applied to only a few

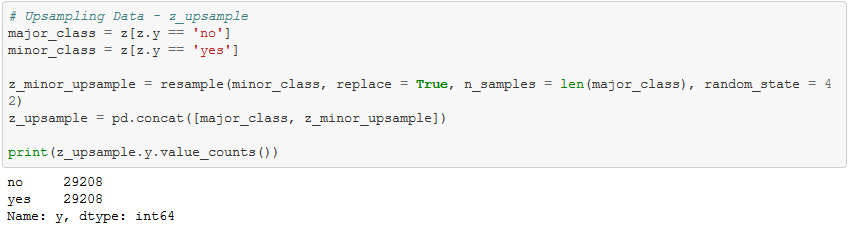


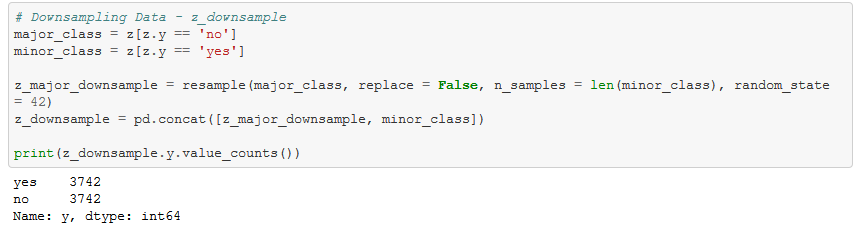
**4. Ready for Machine Learning**

**4.1 Standardization and Normalization**Two popular data scaling methods are normalization and standardization.  
 1. Data Normalization  
 2. Data Standardization

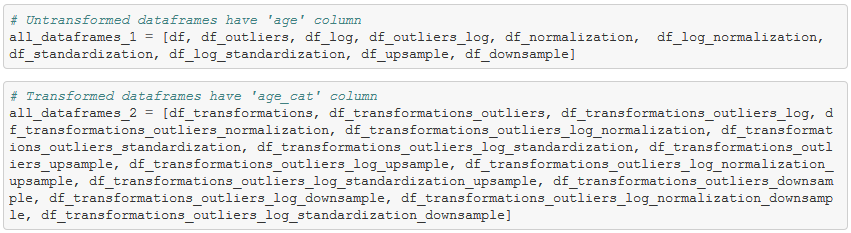
****

**4.2 Upsampling and Downsampling:**

****

****

**4.3 Dummy Variables:**Since each dataframe has different categorical columns, all dataframes are divided into two lists.

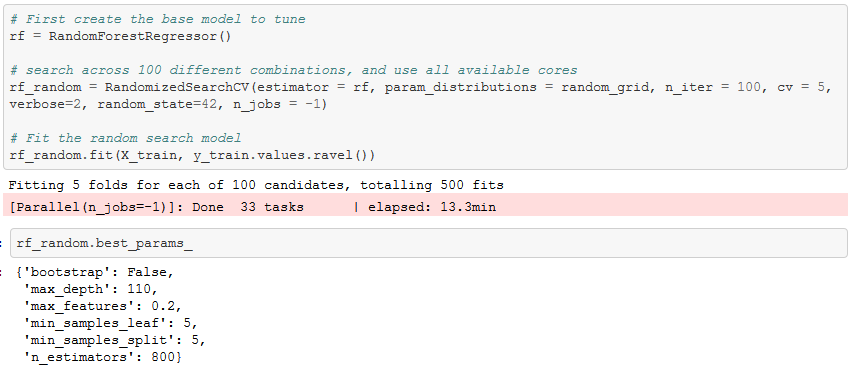


**5. Machine Learning**

**5.1 Random Forests with untransformed data**  
Initially, the data is trained on the base model with no transformations   
 *Accuracy – 0.912, Precision – 0.44, Recall – 0.67, F1 – 0.53, AUC - 518*

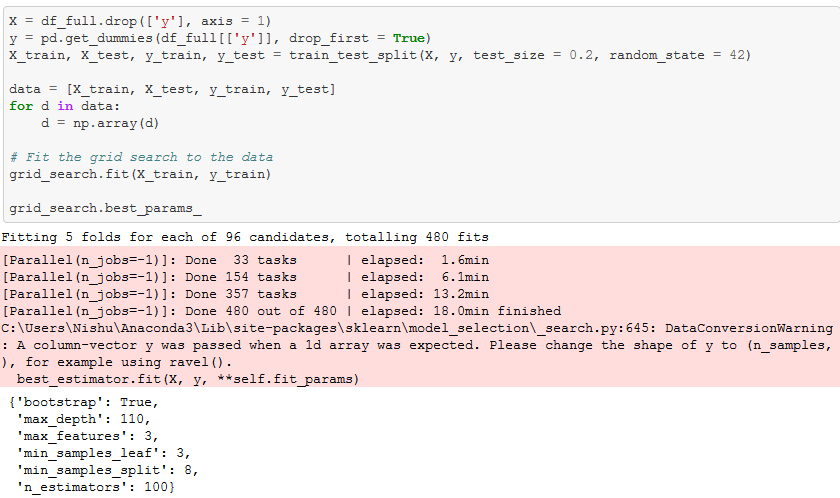
**5.2 Randomized Search CV:**Efficient approach is to narrow our search to evaluate a wide range of values for each hyperparameter.   
 *Bootstrap – False, max\_depth – 110, max\_features – 0.2, min\_samples\_leaf – 5, n\_estimators - 800*

Training the base model with different sets of parameters to find the best set.



**5.3 Grid Search CV:**Using Grid Search CV to pick the best parameters. This gives us an idea where to concentrate our search.   
 *Bootstrap – False, max\_depth – 110, max\_features – 3, min\_samples\_leaf – 3, n\_estimators - 100*

Training the base model with different sets of parameters to find the best set.



**Note:** Since the goal of this project is to minimize False Negatives (How many did we miss), we focus on getting a recall value close to 100% with a less bad precision value

**5.4 Training all the models with grid search CV best parameters:**

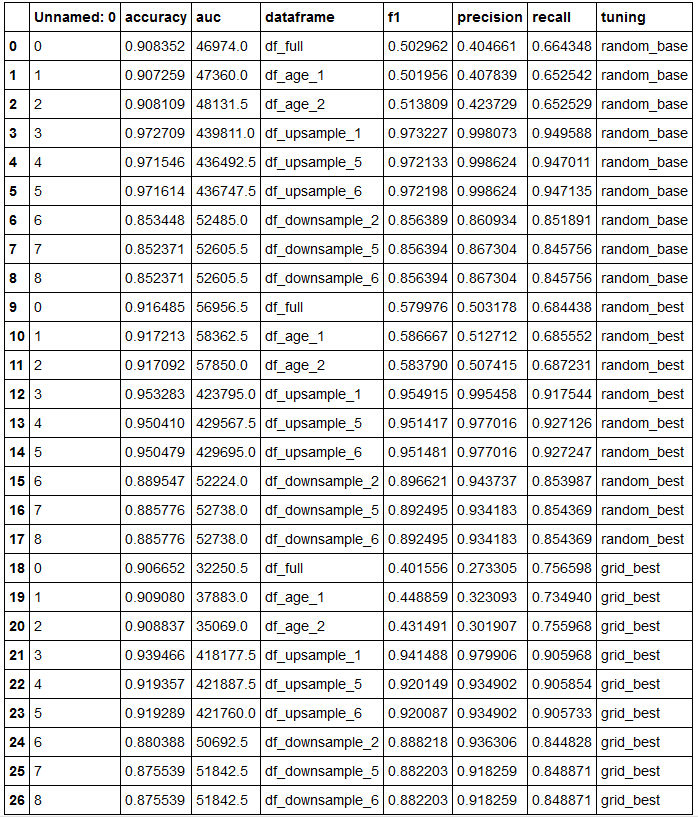
Training all models with best parameters of Grid Search CV (See Section 6.6)

**5.5 Training all models with best parameters of Random Search CV**

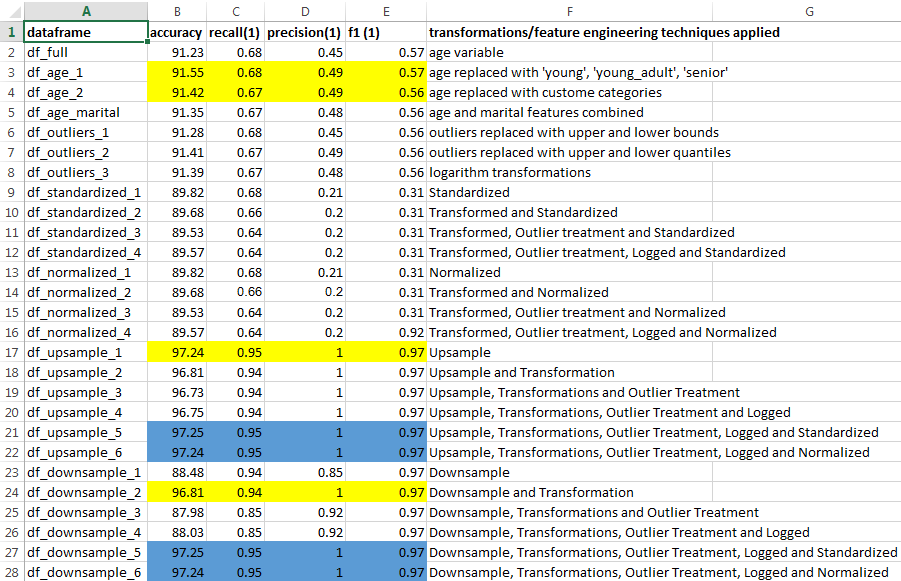
Best parameters of Grid Search CV are chosen over the best parameters Random Search CV considering the computational resources I have. (See Section 6.6)

**5.6 Metric results:**

Below is the table with models are their respective metrics



Here is the list of dataframes tested on the best parameters of Grid Search CV and their respective metrics.



Highlighted are the models with optimal metrics.

**6. Choosing the best model**

From the models highlighted in the above screenshot, df\_age\_2 is the model that yields a better results on the test data.

Other models are not chosen (upsample and downsample) considering the weights each classes are given when the data is either upsampled/downsampled.

**7. Other potential data sets I could use**

The data provided could actually be considered very rich in terms of predicting the client’s behavior for a given campaign. However, given additional data pertaining to client’s financial spending such as income disposal, large credit purchases, demographic of the client.