Article

# Loading Libraries and Dataset

# Exploratory Data Analysis

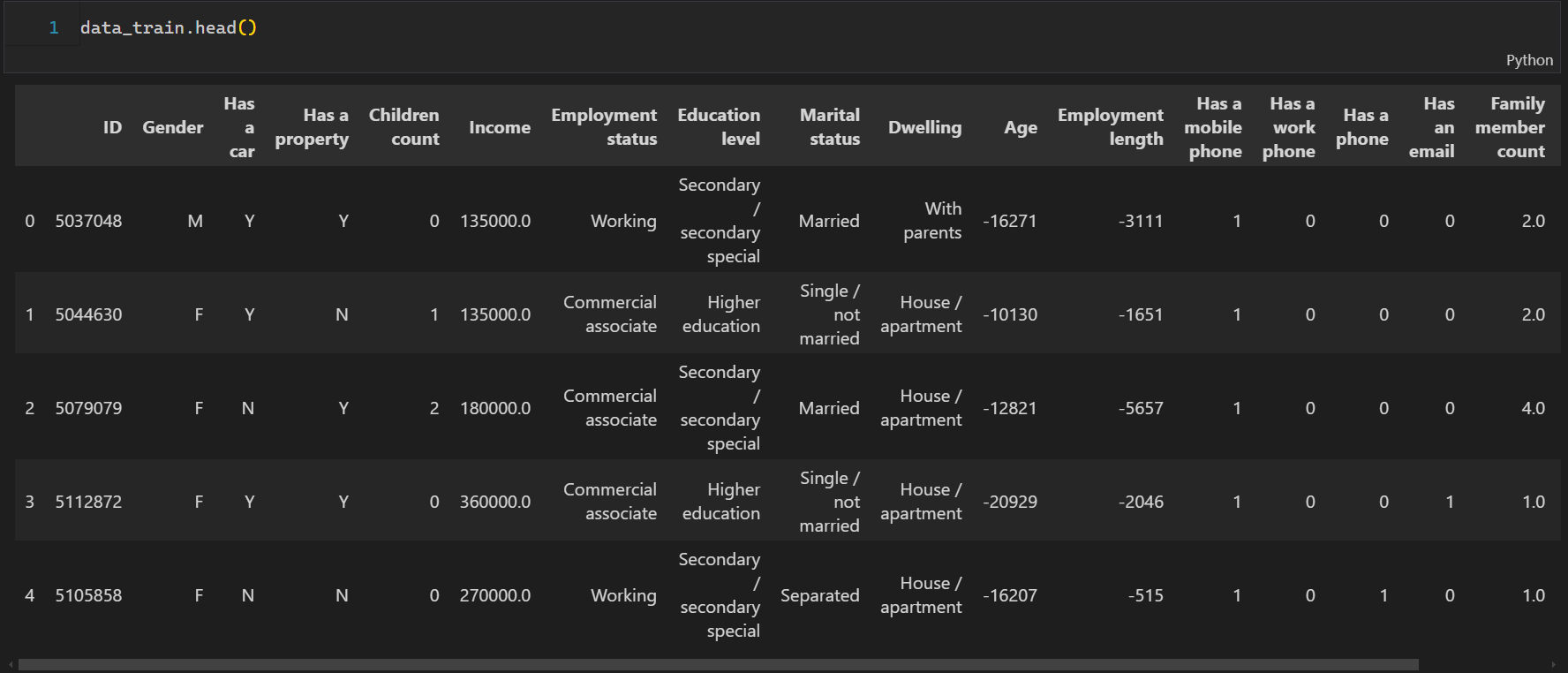
After extracting Null Value count, observe that Job Title is full of them. Thinking logically, a higher up person would tend to earn more, thus can be classified as less riskier than the one earning decently. The point is, any information present in this column is redundant because interesting insights can be drawn from other columns and analysis can be thus made more efficient. It is better to drop this column.

Few observations can be drawn,

1. Average “Children Count” is less than 0.5 but max Children is 19. This, is a potential outlier.
2. Age is negative and reach upto -7705. Possibly the unit is in days. We need to treat this appropriately.
3. Employment Length is same case as Age.
4. Family Member count averages to just above 2 but maximum members exceed 20. This is probably the same outlier mentioned in Children Count.
5. Account age is same as Age, negative in nature. While the former was described in days, Account Age is probably defined in Years and thus would not need to be altered.



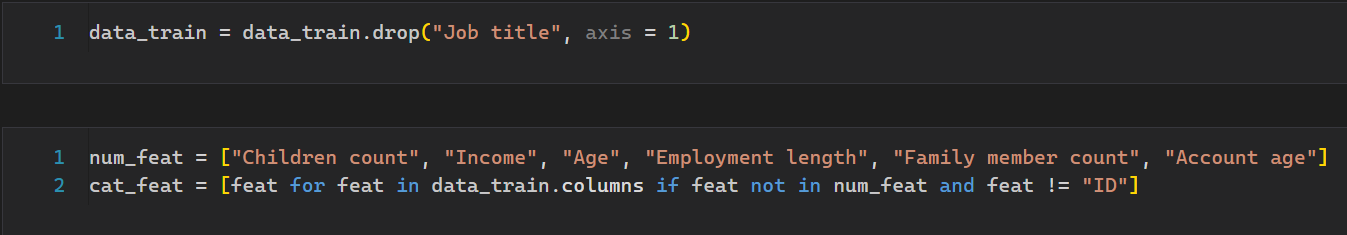
Extracting the data,



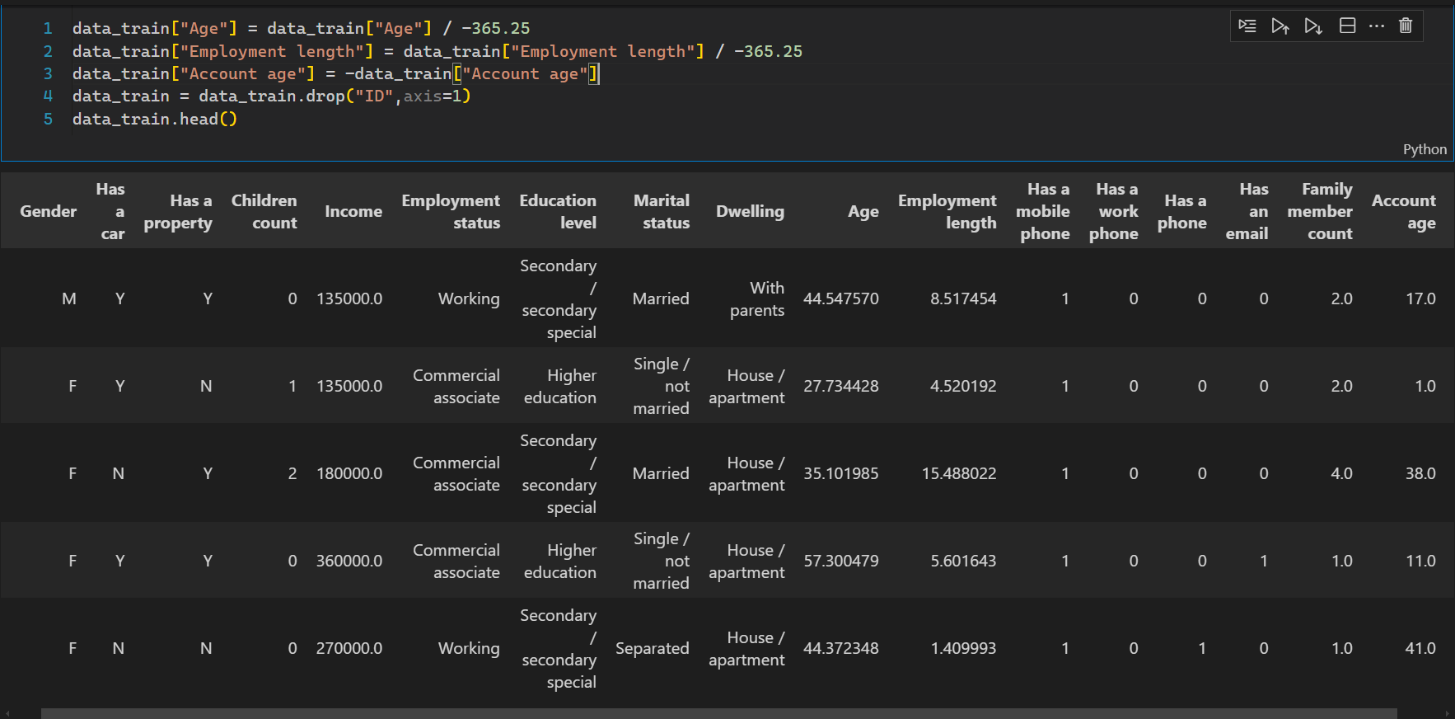
The output is truncated here, but observe that few categorical features are present too!

We need to encode them appropriately. The Encoder present in Sklearn’s Preprocessing library would come handy here.

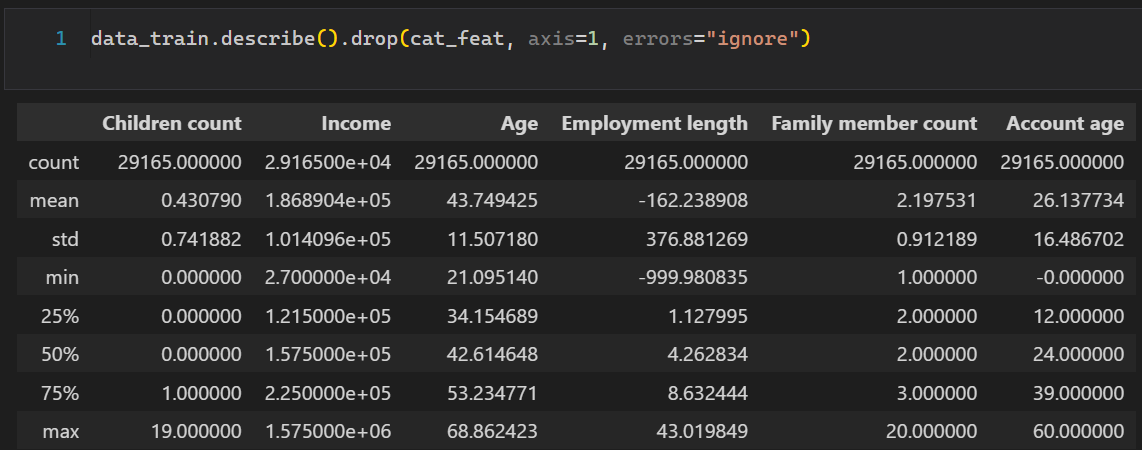
Dividing data into numerical and categorical variables,



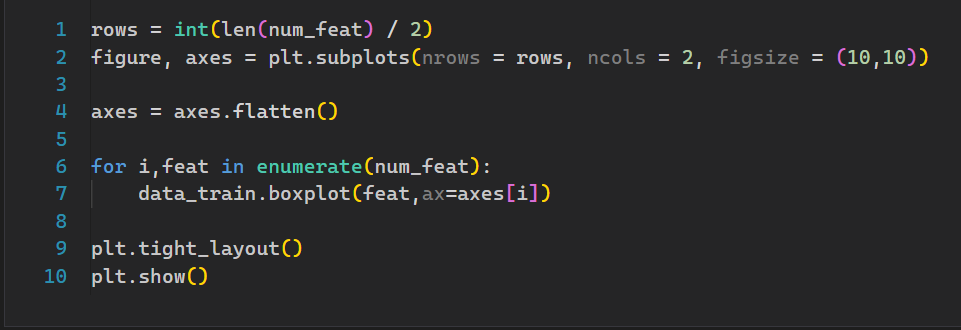
Next up, treating some data anomalies identified earlier,



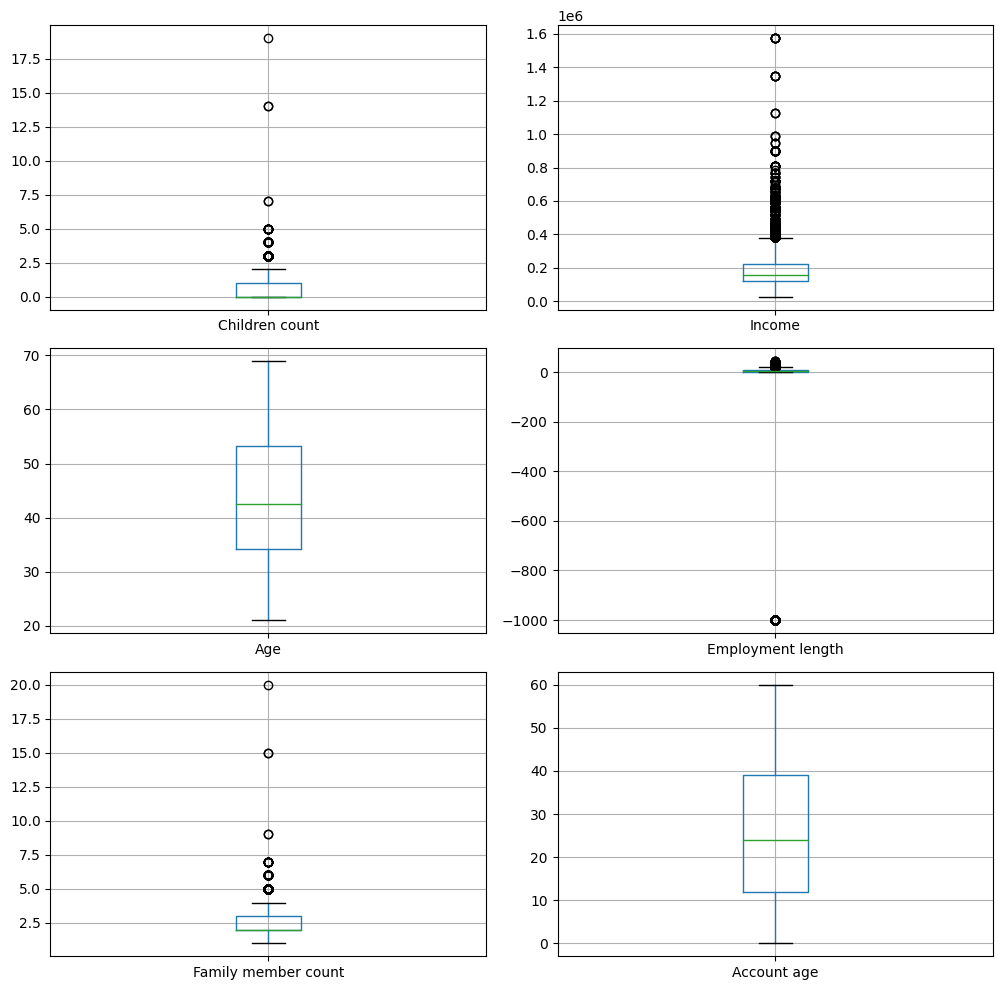
Here is the updated description,



Next up, we need to identify the outliers present in data. The following code snippet conveniently plots the boxplot for various numerical columns present in data.



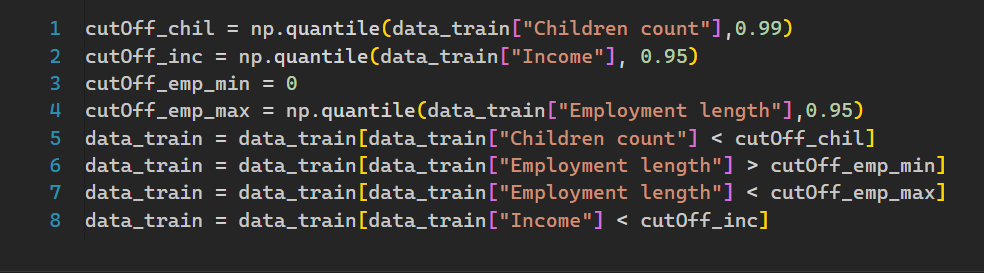
Keep in mind we need to first identify the outliers’ columns, filter them out, define cut off criterion and then treat the column appropriately. This seems a lot but working is quite simple. First up, a simple boxplot can summarize a lot of information in a very concise and visually appealing way. The code snippet provided above does the job. Finally, the plot is given on the next page!



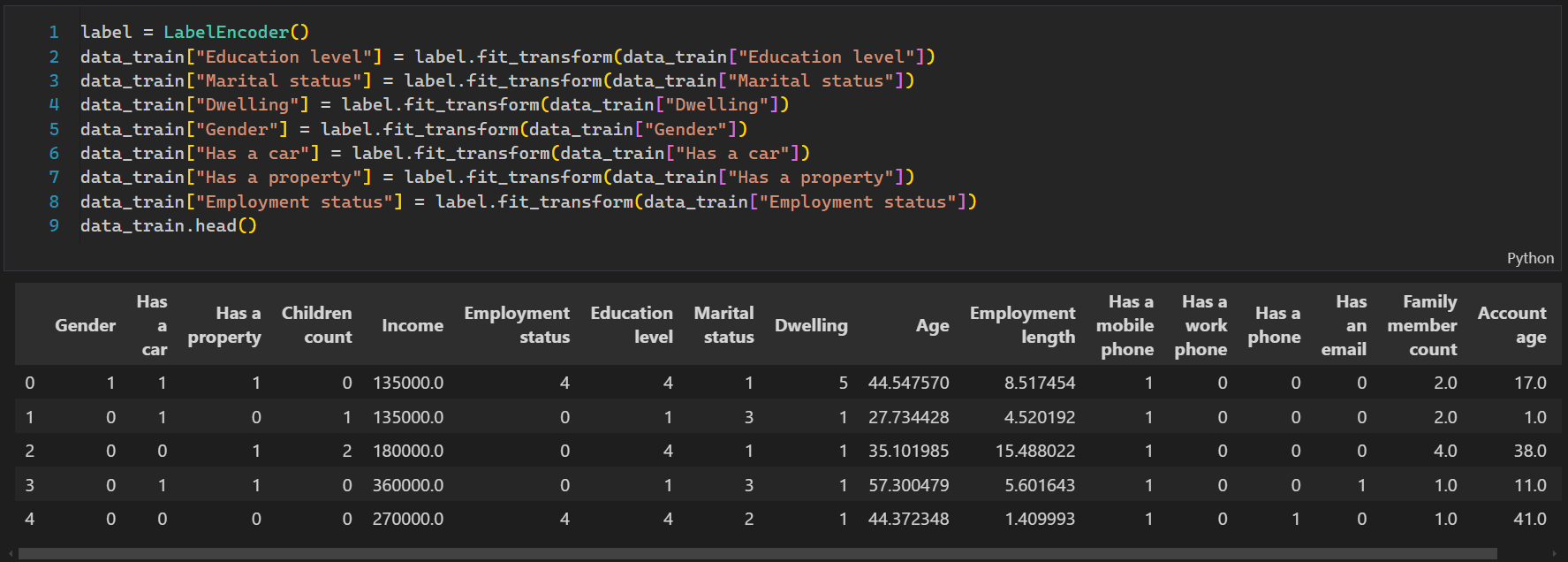
Every black circle here represents a potential outlier. It is upto the data Analyst whether to treat them, alter them, or drop them. In our case, instead of treating them with say, converting them to mean of corresponding column would lead to unintended consequences during model fitting. It’s better to simply remove them as the data is abundant. All that remains is to identify the cut off criterion. Here are a few listed:

1. Observe the Children count consolidates to around 2.5. meaning any value above is an outlier. We can extract the 99th percentile of the this column and drop any tuples above the stated value (2.5).
2. Observe the Income ranges from . That’s ASTONISHING. Instead of setting a very high cut off as in case of Children, we can set it to 95th percentile. The reason being there’s a lot of such values present. We don’t want to loss valuable information.
3. A clear data anomaly is present in Employment length, presence of negative values. Remember we already treated the column for such values and keeping them now would cause problems during model fitting. We should drop these values. Also, few value also exceed majority of values. Observe the black dots present in the upper side of employment length. We also need to treat those values.
4. Observe the Famliy Member count. Highest being 20. This was pretty much clear. We also need to treat such values.

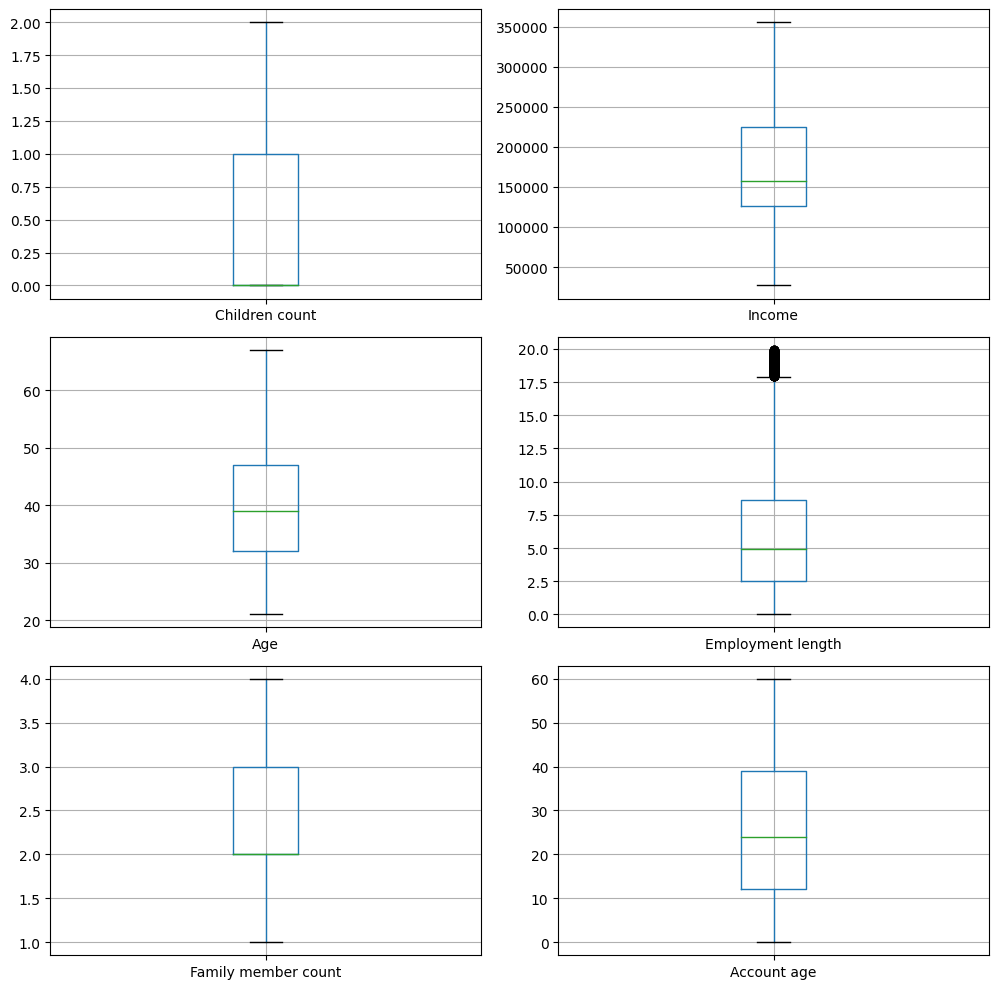
The following code sets the cut off criterion and updated data accordingly,



Next up, we need to encode various categorical features.

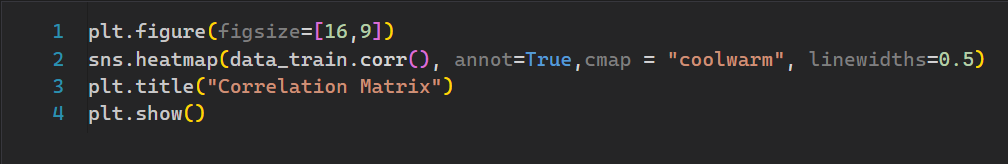


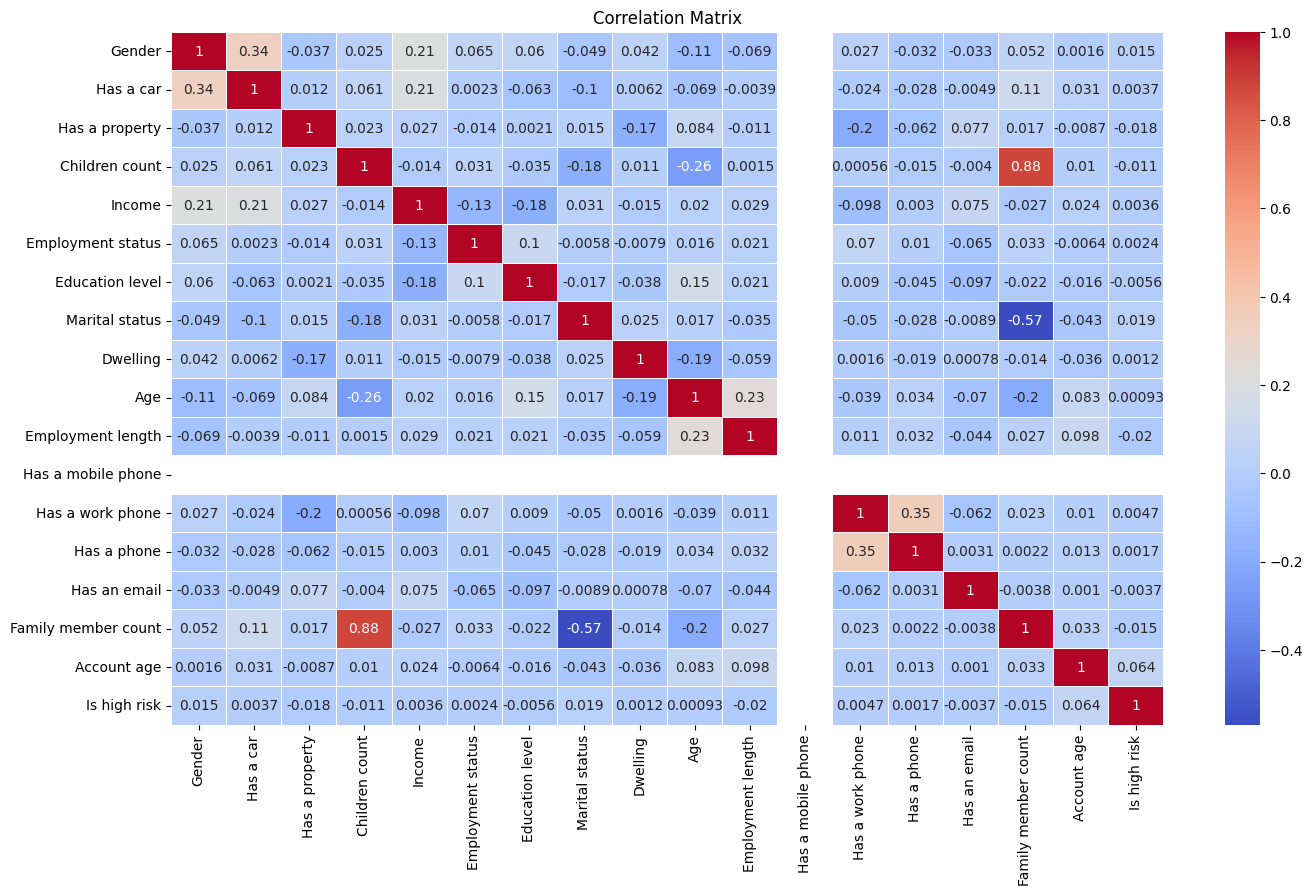
Moving on, there’s the updated Boxplot.



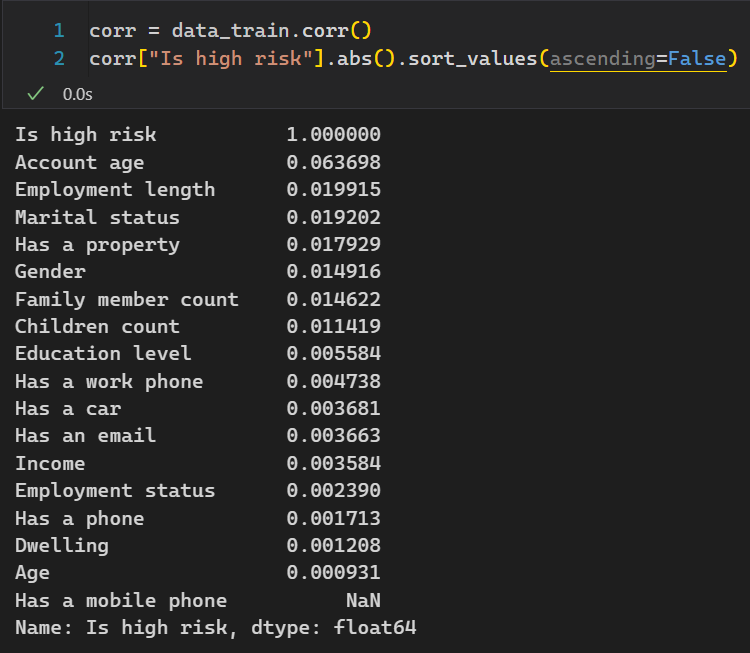
Observe that there are quite a few values still present in Employment length. Instead of dropping them, observe that the error bar far exceeds the body of plot. Meaning, good number of values lies in this range. We can’t drop such values as any unintended data loss is valuable information lost for the model.

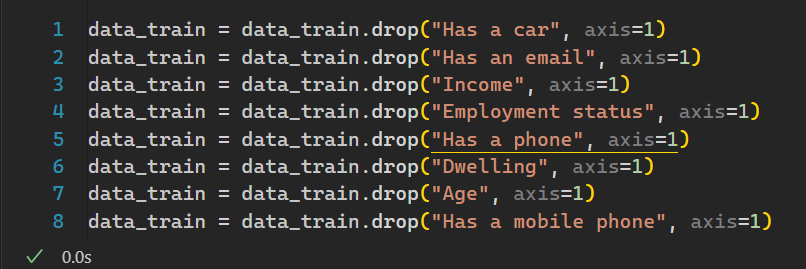
Plotting a correlation matrix to check from important features.



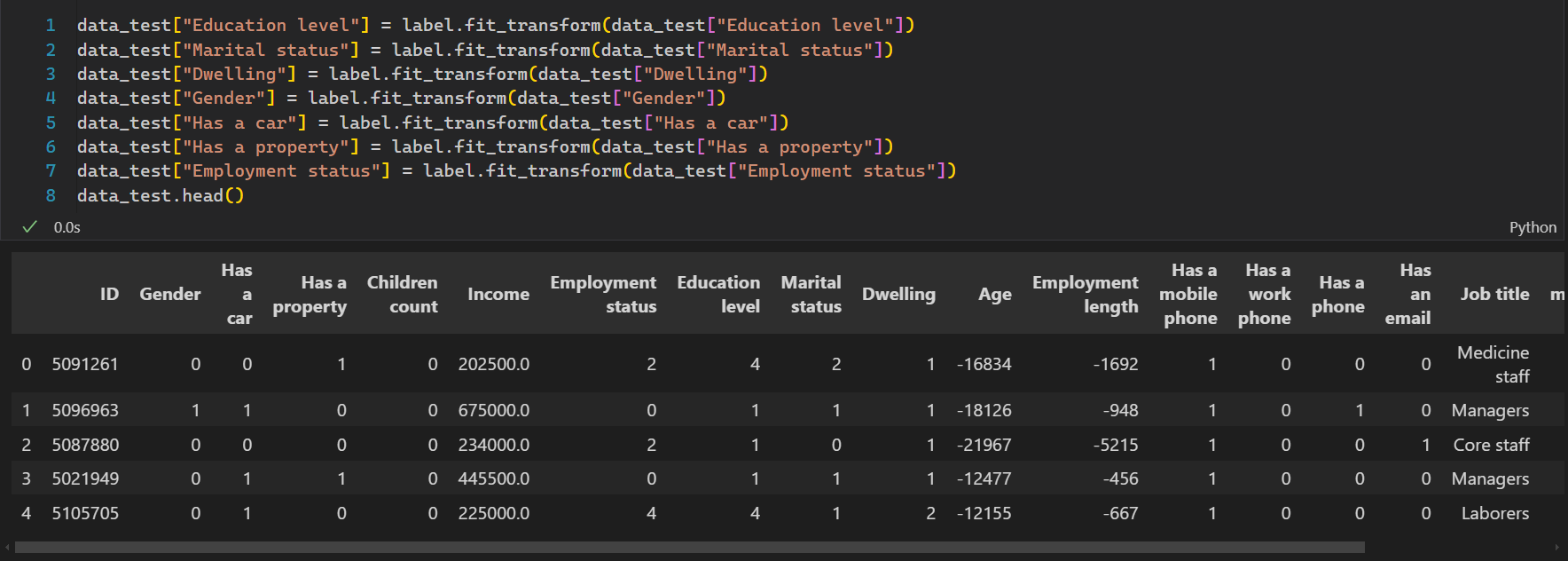


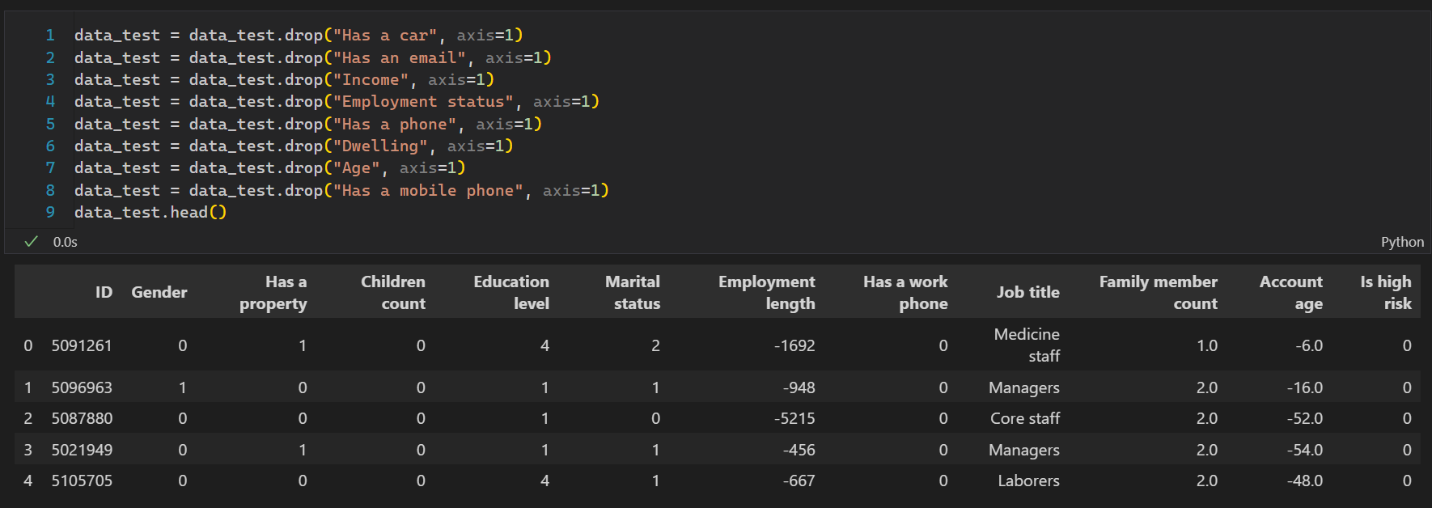
That’s a lot of features! We need to drop the ones’ having very low correlation. Extracting correlation of interest.

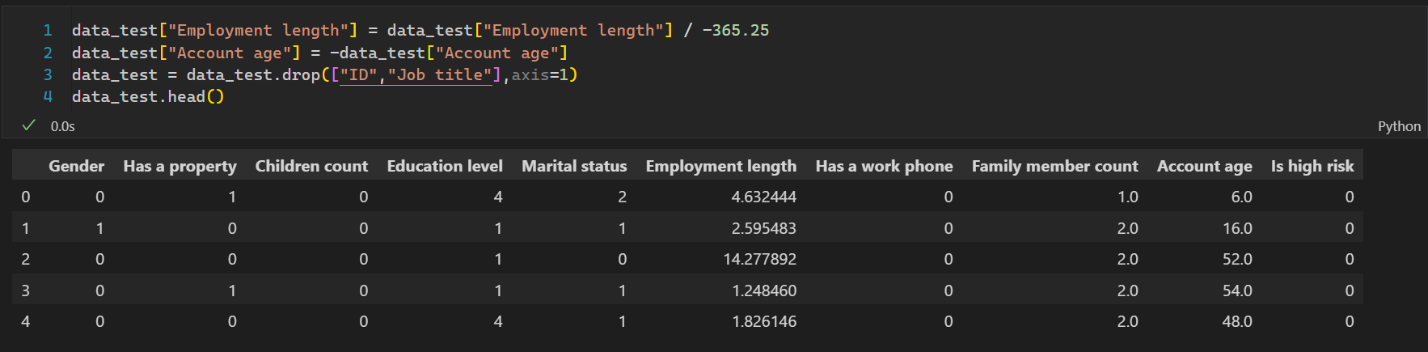
Observe, quite a few show very low correlation with Riskier profiles. Dropping unnecessary features.



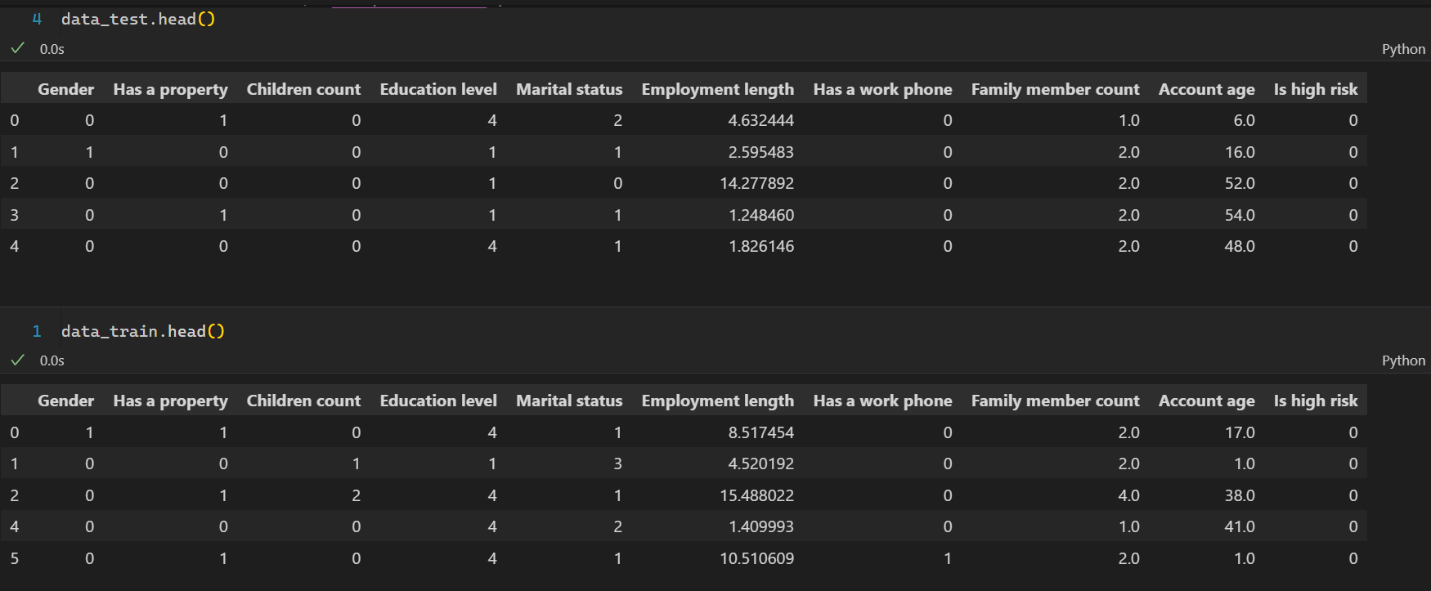
Now, transforming the test data set model fitting.





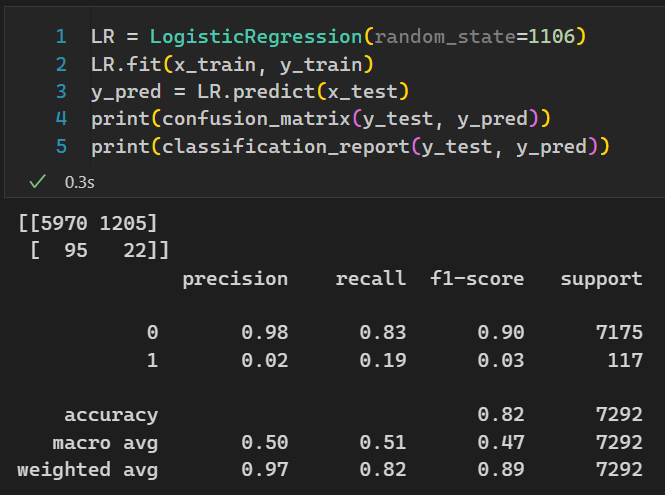


Here is a bird’s eye view of both the testing and training dataset.

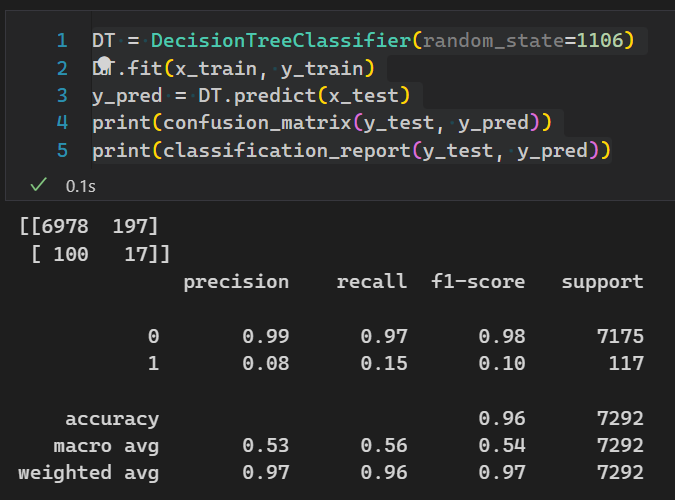


# Model Fitting

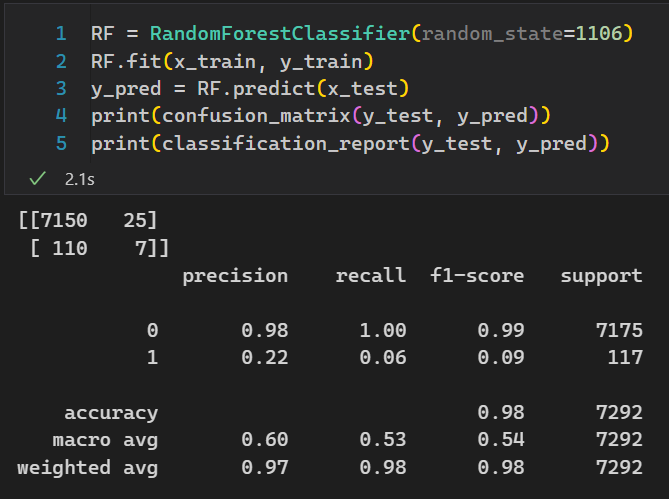
## Logistic Regression



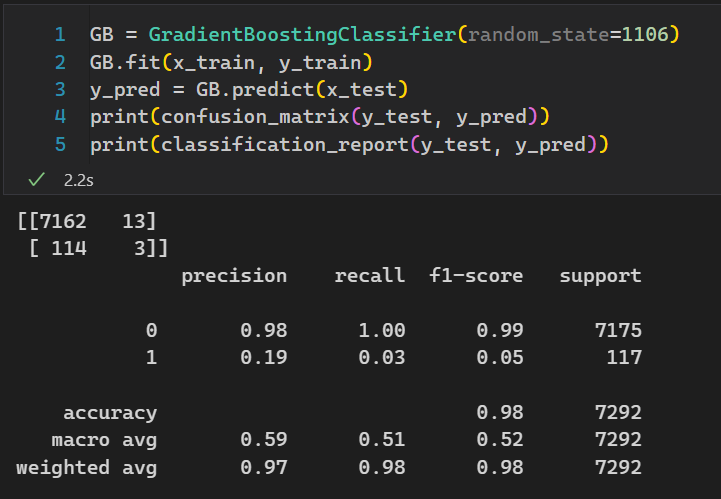
## Decision Forest Classifier



## Random Forest Classifier



## Gradient Boosting Classifier



We need to maximise precision for High risk individuals. Random Forest gives the most optimal result.

Saving the model for future use.

