**How to use it?**

Download the GitHub repository of profiling-tool.

After downloading:

* Go to the directory where profiling-tool-main is downloaded.
* Extract profiling-tool-main
* Then use below-mentioned command in conda prompt to install the profiling-tool package after going to profiling-tool-main directory.



After installing follow below steps:

* Before starting, complete these mandatory steps.
* Open an excel file.
* Open ‘File’ then select ‘Options’.
* In the Excel Options pop up select ‘Trust Center’ then ‘Trust Center Settings’.
* Then select ‘Macro Settings’
* In Macro Settings select ‘Enable VBA macros and check boxes ‘Enable Excel 4.0 macros when VBA macros are enabled’ and ‘Trust access to the VBA project object model’
* Import the class.

A picture containing text, font, screenshot, white

Description automatically generated

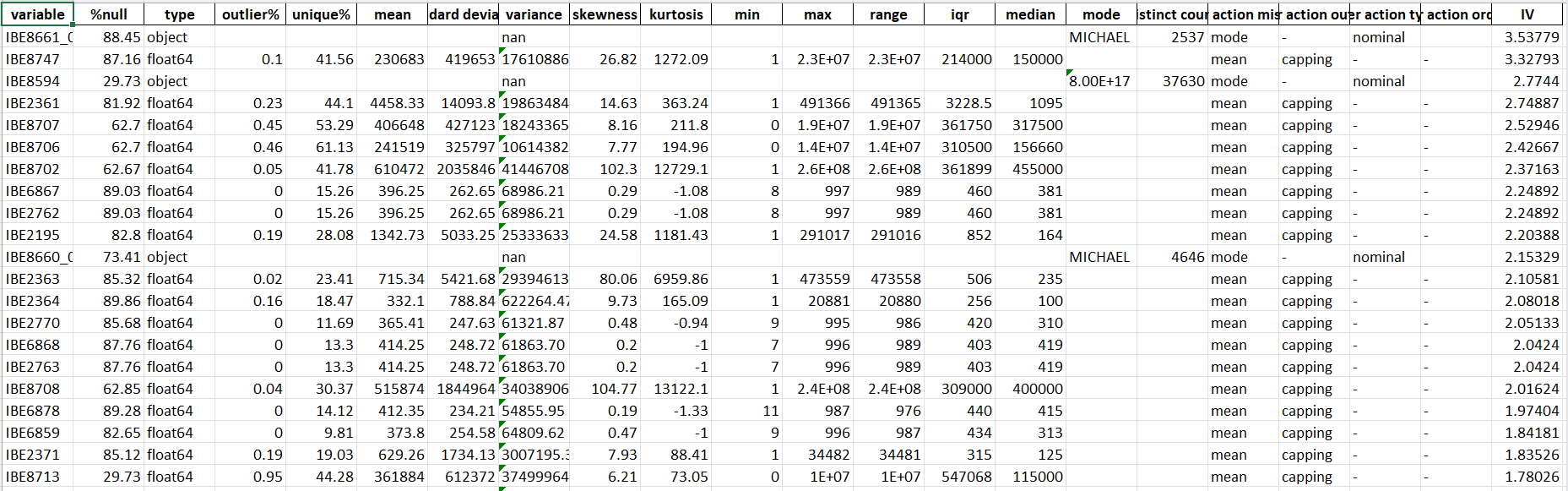
* Run this line.



* When you run the second line,

A picture containing text, font, screenshot, line

Description automatically generatedit generates an excel which contains the summary of all the features in the dataset along with 4 columns dedicated to user specific actions.



*Excel file generated when executed generate\_report () function.*

* For treating missing values, by default ‘mean’ imputation for numerical columns and ‘mode’ imputation for categorical columns are added in ‘user action missing’ column, you can change them if you prefer otherwise. In the similar way to treat outliers in numerical columns we use ‘capping’ as default. And if you don’t want to do any treatments you can keep them blank.
* ‘User action type’ is used to tell if a categorical column is nominal or ordinal by default it is set to nominal. If a column is ordinal, then you must mention the order of values in the column in ‘user action ordinal’. (This action is still in testing phase.)
* Below are all the data preprocessing treatments available:

|  |  |  |
| --- | --- | --- |
| **Missing Value Treatments** | **Outlier Treatments** | **Transformations** |
| Mean Imputation | Remove Outliers | Min-Max Normalization |
| Median Imputation | Capping Outliers | Standardization |
| Mode Imputation | Mean imputation | Log Transformation |
| Linear Interpolation | Median Imputation | Lead Transformation |
| Polynomial Interpolation |  | Lag Transformation |
| Forward Fill |  | Adstock Transformation |
| Backward Fill |  |  |
| Drop Row |  |  |
| Drop Column |  |  |

* After filling all the required treatments run below code.

variable\_profiling() function generates an excel which contains the binned summary of all the features.

The below code is used for performing decile-based binning on numerical columns.

this code does all the transformations and generated an excel which contains all the binned summary of all the columns along with the initial summary of all the features.

For numerical binning we have quantile-based binning and tree-based binning, for quantile-based binning set numerical\_binning as ‘decile’ along with dep\_var to the dependant target variable and maximum number of bins allowed in no\_of\_bins. For tree-based binning set numerical\_binning as ‘tree’ along with dep\_var and no\_of\_bins and an extra parameter depth which represents the depth of the tree.

* To perform tree-based binning on numerical columns run below code.

excel\_report.variable\_profiling(summary\_path='df\_input.csv\_eda\_report\_2.xlsx', numerical\_binning='tree', depth=5, dep\_var='Target', no\_of\_bins=10)

All the features are ordered in the descending order of IV (Information values) values.

* The code also has a built in VBA feature which helps the user to filter only the columns they want to see across all the sheets when performed specific filter in summary sheet in the excel generated and click the ‘Run Macro’ button.

Chart

Description automatically generatedA picture containing graphical user interface

Description automatically generated

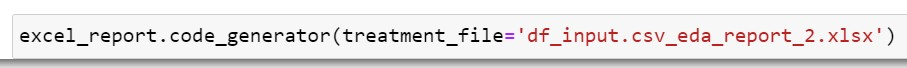
*Binned numerical columns summary Binned categorical columns summary.*

A picture containing table

Description automatically generated

*The summary sheet of the final excel file generated.*

* If you want to generate the code to get the data frame after all the transformations are done run below code.

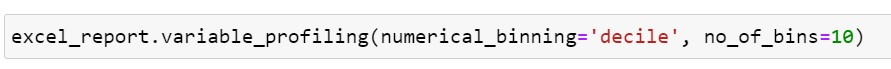
this function will generate the entire code to get the transformed data after performing all the treatments if there are any.

* You can also get the transformed data after all the transformations are performed by running the code below.

A close-up of a sign

Description automatically generated with low confidence

* You can directly go for profiling the data without generating the summary.



* These are all the binning options possible:

**Parameter Descriptions:**



: param null\_percentage\_to\_drop\_column: drop columns which have null percentage above this threshold.

: param dependant\_target\_variable: dependant target variable

excel\_report.variable\_profiling(summary\_path=None, numerical\_binning=None, dep\_var=None, depth=None, no\_of\_bins=None, ordinal\_binning\_type='kprototype', max\_threshold=10, min\_threshold=5, tolerence=2, flag='ignore', min\_cluster\_size=min\_cluster\_size, max\_clusters=max\_clusters):

: param flag: Flag.

: param tolerence: Tolerance.

: param min\_threshold: Minimum threshold.

: param max\_threshold: Maximum threshold.

: param ordinal\_binning\_type: Type of binning function to perform on ordinal categorical column.

: param summary\_path: Generated summary file.

: param numerical\_binning: Type of binning to perform numerical columns.

: param dep\_var: Dependant target variable.

: param depth: Depth of tree in case of tree-based binning.

: param no\_of\_bins: Maximum number of bins allowed.

: param min\_cluster\_size: Minimum size of a cluster.

: param max\_clusters: Maximum number of clusters allowed.

Excel\_report.code\_generator(treatment\_file=None)

: param treatment\_file: Generated summary sheet.