**Lesson Learned**

In this task I need to find the customers who have defaulted on loans. For this analysis I did the following steps: get to know your dataset (explore), pre-processing and EDA.

**Get to know dataset -** I started to analyze dataset in Python for the first time. This step in Python was very similar with R. I used Pandas in order to get to know my data frame. I used pandas.head(), pandas.describe(), pandas.info() functions. These functions showed me the variables within dataset, statistical information, data type of these variables and overall description of this dataset.

**Pre-processing –** In this stepI used Pandas’s functions again.

I binned the age variable. Binning help us to discover patterns, carefully develop algorithms and explain the result to an audience easily.

I deleted ID column since it did not give any value to our dataset.

Pandas recognize both empty cells and “NA” types as missing values. I used dataframe.isnull() function to detect the missing value within dataset.

At the end of this process I wrote all of these changes to a new csv file (dataframe.to\_csv()).

**Exploratory Data Analysis -** EDA was most important step to take before diving into machine learning or statistical modeling because it provides the context needed to develop an appropriate model for the problem at hand and to correctly interpret its results. EDA is using visual and quantitative methods to understand and summarize the dataset without making any assumptions about its contents.

During this step I used Pandas, Matplotlib and Seaborn libraries. Pandas was used for reading the new csv file, Matplotlib and Seaborn used for visualization. I also find the correlation and covariance between the variables on this step.

I got the prettier plots with Seaborn, when I visualized the data.

**Errors and recommendation -** After getting several errors during the analysis I realized that Python is case sensitive, spaces after commas is important, index start from 0 (if the header starting from first raw then on read\_csv() header should be equal to 0).

It is easy to read and understand the errors in Python than it is in R.

After using several types of plots for visualization, I realized that it is not good choice to use line plot while working with the categorical data.

It is better to use Seaborn if you want to summarize your data and still want to show the distribution of your data.

When I wrote the changes into csv file, I use index=False in the function in order to avoid the file has a separate column of indexes.