**Income Qualification.**

Project 101

DESCRIPTION

Identify the level of income qualification needed for the families in Latin America.

**Problem Statement Scenario:**  
Many social programs have a hard time ensuring that the right people are given enough aid. It’s tricky when a program focuses on the poorest segment of the population. This segment of the population can’t provide the necessary income and expense records to prove that they qualify.

In Latin America, a popular method called Proxy Means Test (PMT) uses an algorithm to verify income qualification. With PMT, agencies use a model that considers a family’s observable household attributes like the material of their walls and ceiling or the assets found in their homes to  
classify them and predict their level of need.

While this is an improvement, accuracy remains a problem as the region’s population grows and poverty declines.

The Inter-American Development Bank (IDB)believes that new methods beyond traditional econometrics, based on a dataset of Costa Rican household characteristics, might help improve PMT’s performance.  
Following actions should be performed:

* Understand the type of data.
* *Write a function* print\_various\_data\_info with input parameter as DataFrame. This function will *print various info of DataFrame (How many entries, how many columns, column name(s), how many not null rows for each column, DataType of each column, Shape of data, Null count for each columns.*
* Print the first 10 rows. Print the last 10 rows,
* *Write a function print\_numerical\_columns with input parameter as DataFrame to print numerical columns (Int/Float). Return column list.*
* *Write a function* get\_int\_float\_dtype\_null\_column\_list

*with input parameter as DataFrame to print numerical columns (Int/Float) containing Null value. Return column list.*

* *Write a function* fill\_int\_float\_dtype\_null\_cells\_data\_with\_mean with parameters as DataFrame and column\_list *to fill Null cells of int\_or\_float column(s) with mean() value of the corresponding column.*
* *Checking if there any Duplicate rows*
* *Write a function* get\_object\_dtype\_column\_list with parameter as DataFrame *to get the list of all Object Dtype Column(s). Return column list.*
* *Analyzing the value\_count of various types of 'ocean\_proximity' to see if it has limited set of data for encoding purpose – (find all possible values in this column) – can skip in this assignment*
* *Write a function* get\_category\_column\_list with parameters as DataFrame and object\_column\_list *to get Object Dtype categorical column(s) having <= 10 diff type of data. Return* category\_column\_list an  
  d category\_column\_value\_list. Second one is list of list.
* Write a function apply\_category\_column\_encoding with parameters as DataFrame, category\_column\_list, category\_column\_value\_list and create additional columns and drop the original column. Additional column will contain 1 or 0 depending upon the corresponding column and categorical value. – housing file ; add each item of such columns to create a new column and it should be in binary format .
* Check if there are any biases in your dataset. – (there may be varying values; check if data is equally distributed and stuff- know functions about random over-sampling )
* Check whether all members of the house have the same poverty level. ( group by family idhogar and check their level poverty levels)
* Check if there is a house without a family head. – use parentesco1
* Set poverty level of the members and the head of the house within a family.
* Remove null value rows of the target variable.
* Merge the two DataSets

**Steps Taken to Solve**

* Downloaded latest version of python 3.9.7 (64 bit: Anaconda), Powershell 7.2.1 win x-64 msi, Numpy library, Pandas library
* Open Jupyter notebook on Anaconda (import pandas as pd and import numpy as np)

**Understanding the data**

# Function to print various info of DataFrame: e.g., .info(), .shape, Null count of various columns in the DataFrame

def print\_various\_data\_info(df\_data):

#DataFrame.info() method prints information about a DataFrame including the index, dtype & columns,

# count of non-null values. This will indicate the column(s) having null values.

print('Various Info about Data: \n')

df\_data.info()

print('\n Shape of Data: \n', df\_data.shape)

print('\n Null count for various columns in Data: \n', df\_data.isna().sum())

* + Facing issue in printing the integer type list