kaggle

Approach to the application of Data Science and ML as key tools for the construction of public policies in Colombia

Claudia Reyes

Background

GDG Cloud

Women Techmakers

- Electronics engineer with more than 10 years of experience in the area of telecommunications and network security turned data engineer and ML enthusiast.
- From Bogotá Colombia
- Director GDG Cloud Sabana Cundinamarca
- Women Tech Makers Ambassador

Project presentation

Project Summary

This project establishes the study and analysis of various socio-economic variables that influence and determine the generation of state social policies in Colombia.

Throughout this project, the state of the art, the problem statement, the background, the justification of the project, as well as the theoretical and methodological framework which is based on CRISP-DM and implements the analysis corresponding to the selected data set are developed.

The corresponding descriptive and predictive models are generated through data science and machine learning tools to generate conclusions and proposals that can become a basic instrument for the generation of policies that have a real and profound impact on poor and vulnerable populations in Colombia.

Project Summary

The analysis implemented in this paper is based on the Gran Encuesta Integrada de Hogares (GEIH) 2021 implemented by the National Administrative Department of Statistics (DANE).

This is a survey that requests information on people's employment conditions, socioeconomic situation, as well as the general characteristics of the population.

The data set was analyzed for the five main cities Bogotá, Medellín, Barranquilla, Bucaramanga, and Cali.

In addition, the art study of the project was based on several master documents of socio-economic studies which are described in the bibliography.

Learning path

- **Pandas**

- **Data Cleaning**

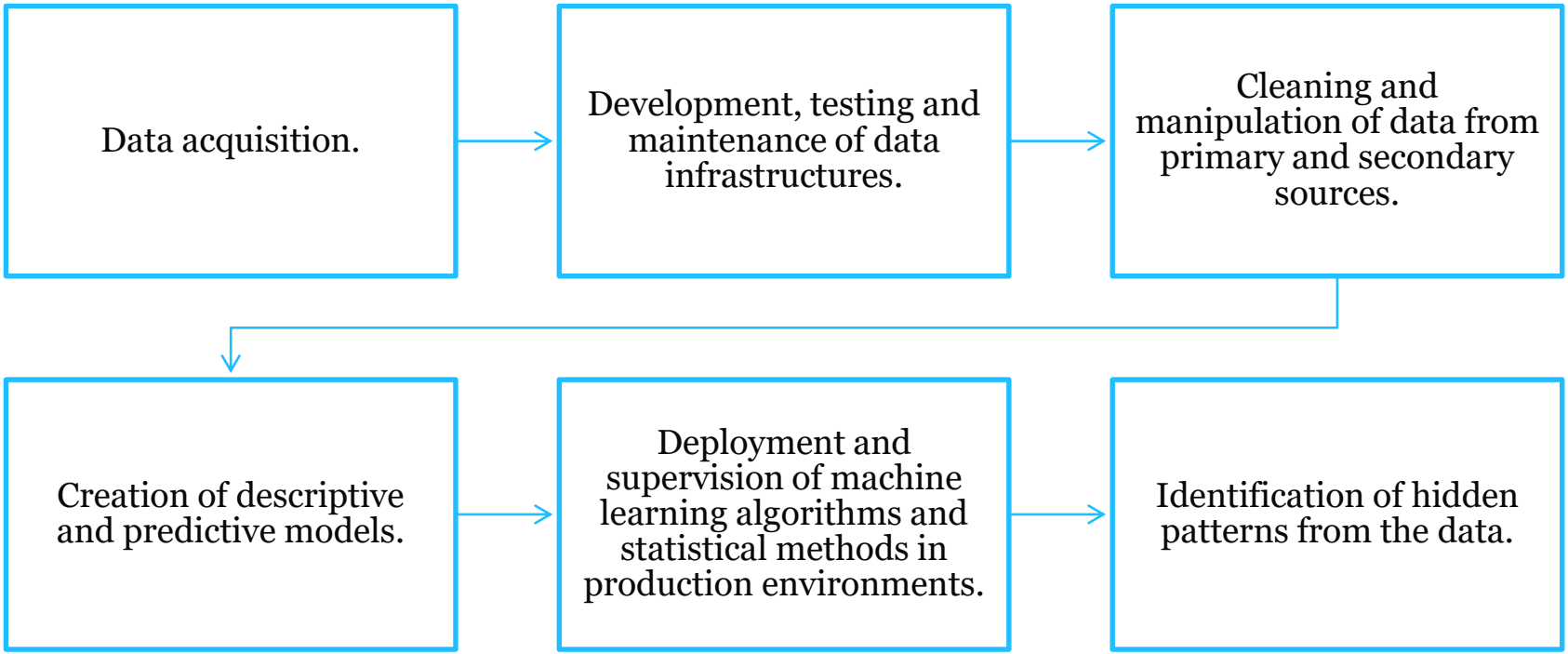
- **Data Visualization**

- **Intro to Machine Learning**

- **Intermediate Machine Learning**

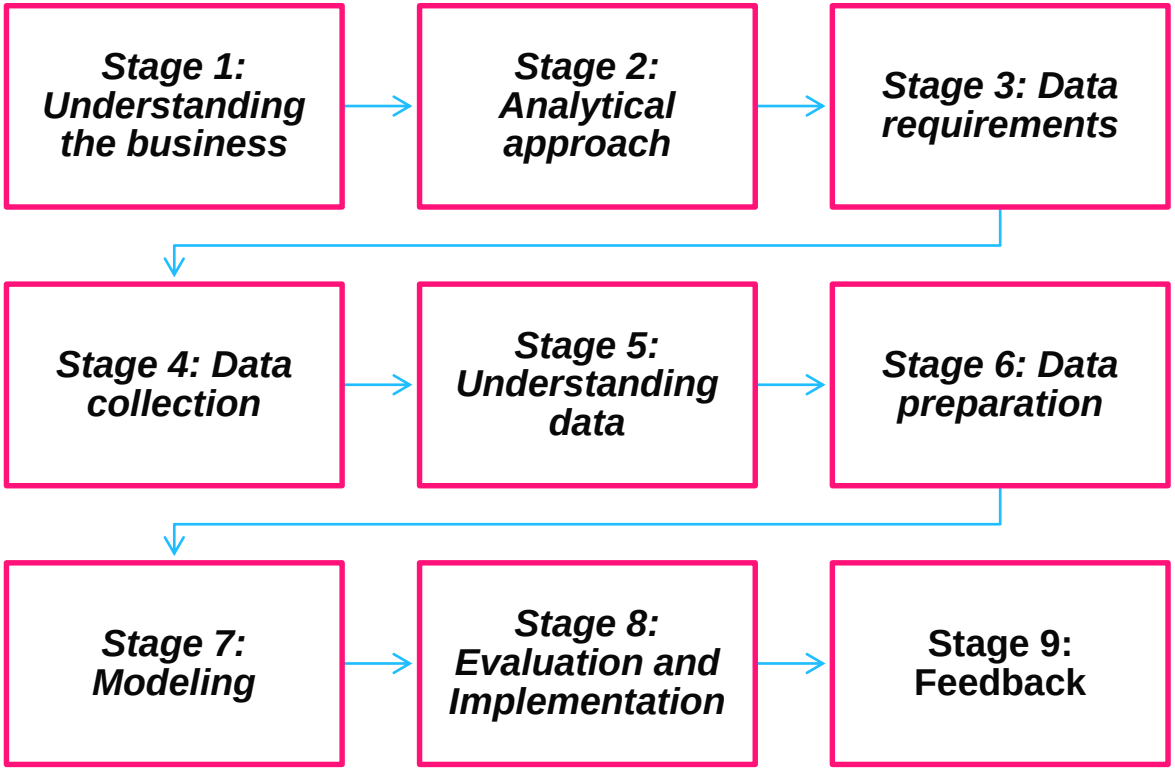
- **Intro to Deep Learning**

Data Science Topic(s) Applied



"Fundamental Methodology for Data Science"
IBM -CRISP-DM

"Fundamental Methodology for Data Science" IBM -CRISP-DM



Stage 1: Understanding the business. Problem definition

- **The high level of inequality in Colombia is a fundamental constraint to economic growth and social progress. The country has one of the highest levels of income inequality in the world; the second highest among 18 countries in Latin America and the Caribbean (LAC), and the highest among all OECD countries (World Bank, 2021).**
- **In this context, the Colombian state has generated targeting elements such as Socio-Economic Stratification and Sisbén in order to generate public policies conducive to mitigating poverty and generating social mobility for its most vulnerable population.**
- **Throughout this study, the aim is to analyze the relevance of various variables involved in public decision-making and the generation of programs whose main objective is to impact the neediest population.**

Stage 1: Understanding the business
Aims of the project

General Objective

- **To analyze the relevance and correlation of targeting tools such as socio-economic stratification with the public policies generated in the social and economic spheres by the Colombian state.**

Specific objectives:

- **To create a base document that allows the application of emerging technologies in this case data science tools and ML that will serve as a guide in subsequent studies and projects involving public policies of the Colombian state.**
- **Serve as a reference document in its technical component, for the implementation of training programs in the area of data science and ML within the programs "data science for all" and "All to code" initiatives of the GDG Sabana Cundinamarca by Google group of which I am the director.**

Stage 2: Analytical approach

Throughout this work, we will apply data science techniques corresponding to descriptive statistics as well as visualization techniques to understand the content of the data.

As well as the introduction of predictive analysis through ML tools

Stage 3: Data requirements

The data collected corresponds to:

Socio-economic information for Colombia updated to 2021 by the World Bank.

DANE (Colombian National Department of Statistics).

The Economic Commission for Latin America (ECLAC).

the Kaggle portal.

The Colombian government's open data portal.

All datasets will be worked on in CSV format to which different tools of the emerging technologies listed above will be applied.

This will be implemented in the Visual Studio Code editor.

Stage 4: Data collection

The primary dataset information for this phase of the project is provided below, stating the file_name in csv format, the source from which it was taken, and a brief description of the dataset provided by the dataset creators. General information about the data is also provided.

This procedure was implemented for each of the seven (7) datasets.

1. Large Integrated Household Survey - GEIH (2021) DANE

File_name: cityWorkCol.csv

Source1:
http://microdatos.dane.gov.co/index.php/catalog/701/get_microdata

Data Information:

Owner of the dataset: National Administrative Department of Statistics (DANE)

Language: Spanish

Geographical coverage: Departmental Capital City

Update Frequency: Quarterly

Issue Date (yyyy-mm-dd): 2023-01-01

General Description: Large Integrated Household Survey - GEIH - 2021 elaborated by DANE. In this document, you will find the historical evolution of the measurement of the labor market in Colombia and the main technical characteristics of the Large Integrated Household Survey.

Data Information:

Owner of the dataset: National Administrative Department of Statistics (DANE)

Language: Spanish

Geographical coverage: Departmental Capital City

Update Frequency: Quarterly

Issue Date (yyyy-mm-dd): 2023-01-01

Stage 5: Understanding data

In this stage, tools are applied that allow in-depth knowledge of the data being analyzed. Likewise, descriptive statistics are implemented.

Loading modules and libraries: The following is the import of the modules and libraries required for the implementation of this project.

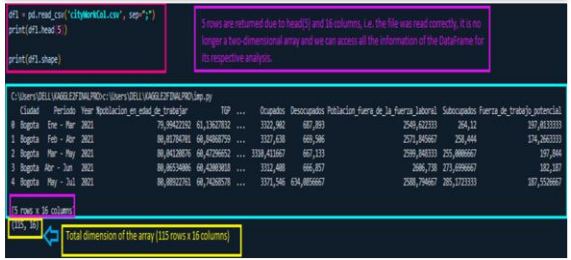
```
######## Computing modules
import numpy as np
import pandas as pd
import scipy
import mlxtend
import datetime
import math

# # for Box-Cox Transformation
from scipy import stats
from scipy.stats import norm

# # for min_max scaling
from mlxtend.preprocessing import minmax_scaling
```

```
########## Plotting  modules
import matplotlib as mpl
import matplotlib.pyplot as plt # For Data Visualization

import seaborn as sns #For Data Visualization
import seaborn.objects as so

import plotly
# Using plotly.express
import plotly.express as px
import dash
#import dash_core_components as dcc
from dash import dcc
#import dash_html_components as html
from dash import html
import plotly.graph_objects as go
import plotly.io as pio
```

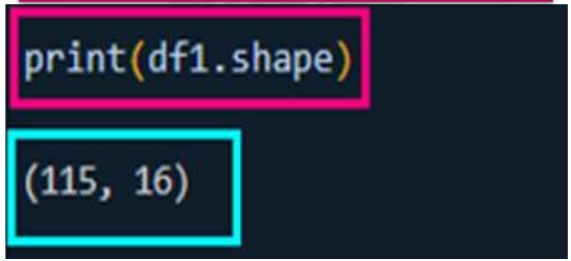
Stage 5: Understanding data

1. The visualization of the csv file is presented.

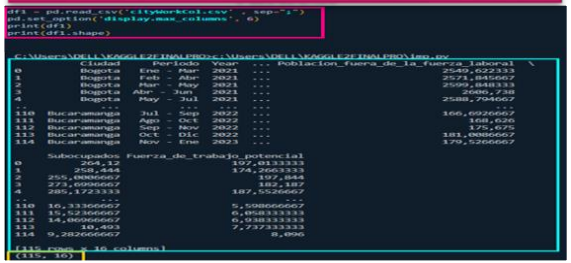
2. Dataframe creation.



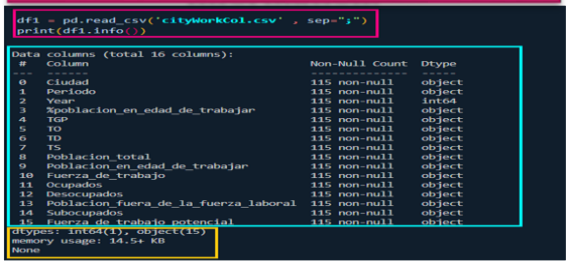
3. Determining the dimensions of the DataFrame.



4. Specific selection of Dataframe values.



5. Obtaining general data information.



Stage 5: Understanding data

6. Checking the data type

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
print(df1.dtypes)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Ciudad                                object
Periodo                               object
Year                                   int64
%poblacion_en_edad_de_trabajar        object
TGP                                   object
TO                                    object
TD                                    object
TS                                    object
Poblacion_total                       object
Poblacion_en_edad_de_trabajar         object
Fuerza_de_trabajo                     object
Ocupados                              object
Desocupados                           object
Poblacion_fuera_de_la_fuerza_laboral  object
Subocupados                           object
Fuerza_de_trabajo_potencial           object
dtype: object
```

7.Request a list of unique values

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
print(df1.Ciudad.unique())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
['Bogota' 'Medellin' 'Cali' 'Barranquilla' 'Bucaramanga']
```

8. List of unique values and their frequency.

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
print(df1.Ciudad.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Bogota          23
Medellin        23
Cali            23
Barranquilla    23
Bucaramanga     23
Name: Ciudad, dtype: int64
```

9. Descriptive summary of the DataFrame

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
print(df1.describe())
print(df1.describe(include = object))
```

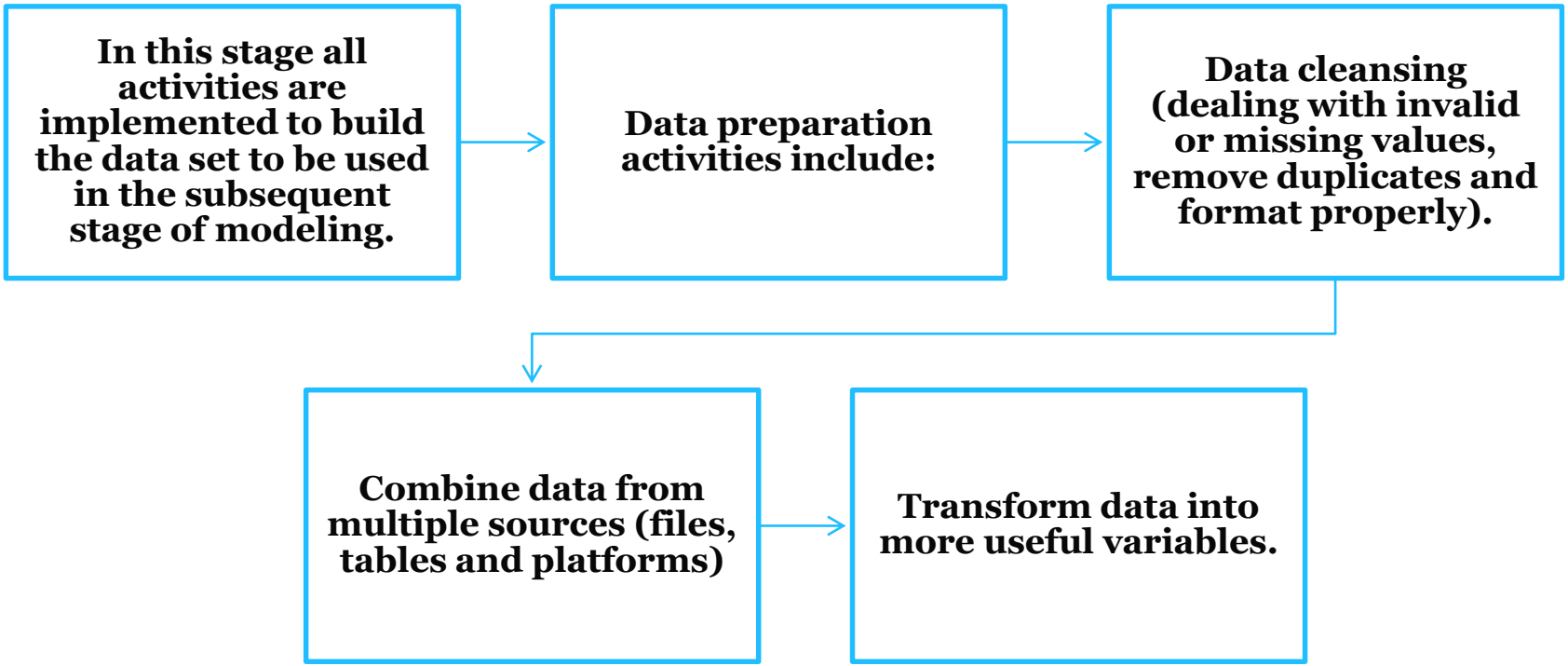
```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
              Year
count   115.000000
mean   2021.565217
std       0.579329
min    2021.000000
25%    2021.000000
50%    2022.000000
75%    2022.000000
max    2023.000000
```

Corresponding analysis of numerical data

Analysis corresponding to non-numeric data

```
        Ciudad  Periodo %poblacion_en_edad_de_trabajar   TGP    TO  ... Ocupados Desocupados Poblacion_fuera_de_la_fuerza_laboral Subocupados Fuerza_de_trabajo_potencial
count      115      115                            115   115   115  ...       115         115                                  115          115                          115
unique       5       13                             97   110   113  ...       115         115                                  115          114                          115
top     Bogota  Ene - Mar                          81,3  64,6  59,0  ...  3322,902     687,893                           2549,622333  68,78533333                197,0133333
freq        23       10                              6     3     2  ...         1           1                                    1            2                            1

[4 rows x 15 columns]
```

Stage 6: Data preparation



Stage 6: Data preparation

Identification and deletion of NaN type data

```
df1 = pd.read_csv('cityWorkCol.csv')
print(df1.notnull().sum())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning.py
Ciudad;Periodo;Year;%poblacion_en_edad_de_trabajar ;TGP;TO;TD;TS;Poblacion_total;Poblacion_en_edad_de_trabajar;
Fuerza_de_trabajo ;Ocupados;Desocupados;Poblacion_fuera_de_la_fuerza_laboral;Sub
ocupados;Fuerza_de_trabajo_potencial  115
dtype: int64
```

115 notnull data

Identification and elimination of duplicate data

```
df2 = pd.read_csv('GEIH.csv' , dtype='unicode')
df2_dup =df2.duplicated().sum()
print(df2_dup)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imple1.py
0
```

Data type conversion

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
df1['TGP'] = df1['TGP'].str.replace(',', '').astype(float)
df1["TGP"] = pd.to_numeric(df1["TGP"], downcast="float")
print(df1.TGP.dtype)
print(df1.TGP.mean())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning.py
float32
4694875600.0
```

Removal of special characters and whitespaces

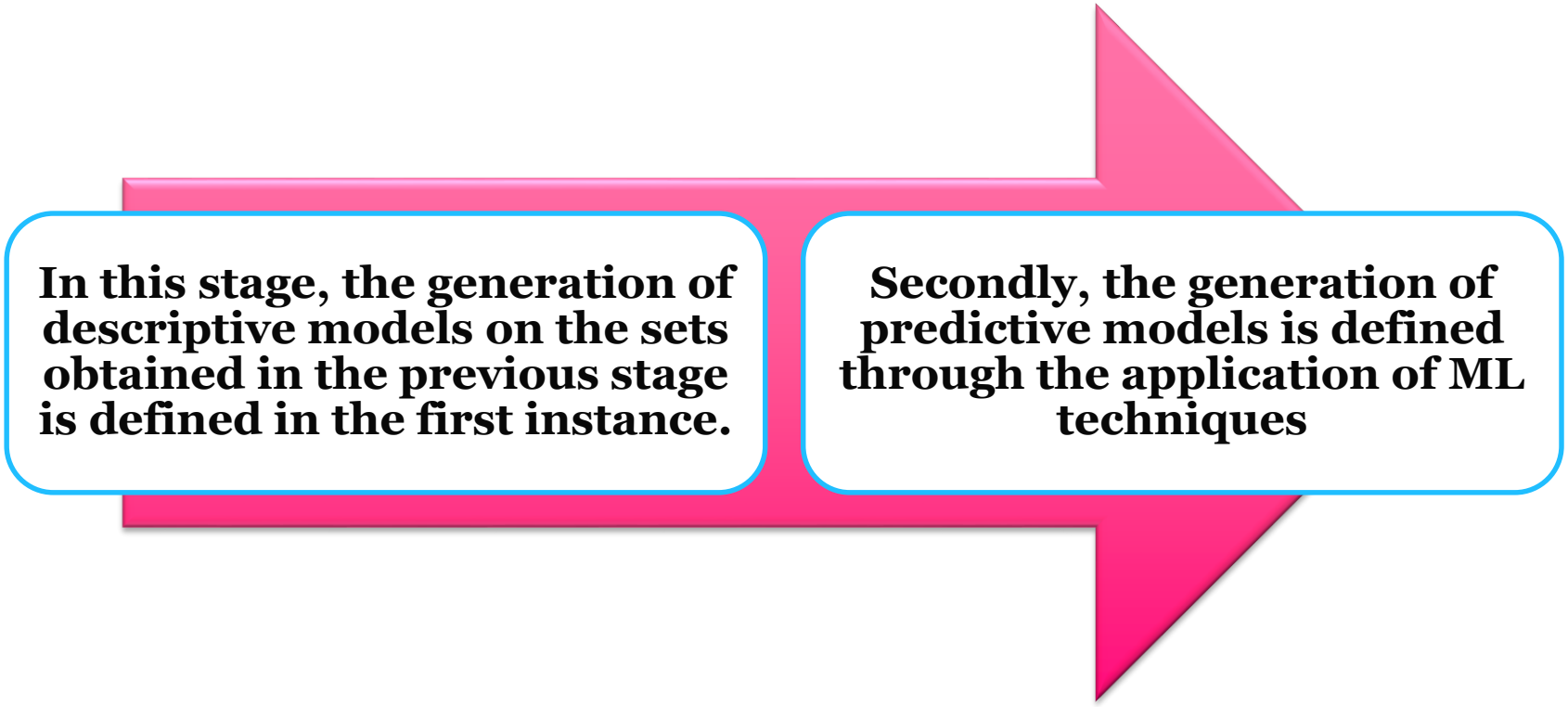
```
df5 = pd.read_csv('PovertyMechanismsofchange.csv', sep=";")
df5.columns = df5.columns.str.strip()
print(df5.columns)

df5["Línea de pobreza"].str.split()
df5["Línea de pobreza"].str.split(expand=True)

LineP = df5["Línea de pobreza"].str.split(expand=True)
LineP.columns = ['Clasificaion', 'IngresoMin','IngresoMax','Año_Medicion', 'PurPoPa']
df = pd.concat([df5, LineP], axis=1)
print(df)

df.columns = df.columns.str.strip()
df["IngresoMin"] = df["IngresoMin"].apply(lambda x: x.replace("$",""))
df["IngresoMax"] = df["IngresoMax"].apply(lambda x: x.replace("$",""))
print(df)
```

Stage 6: Data preparation

Separation of data into individual columns for further analysis.

```
df5 = pd.read_csv('PovertyMechanismsofchange.csv', sep=";")
df5.columns = df5.columns.str.strip()
print(df5.columns)

df5["Línea de pobreza"].str.split()
df5["Línea de pobreza"].str.split(expand=True)

LineP = df5["Línea de pobreza"].str.split(expand=True)
LineP.columns = ['Clasificaion', 'IngresoMin','IngresoMax','Año_Medicion', 'PurPoPa']
df = pd.concat([df5, LineP], axis=1)
print(df)

df.columns = df.columns.str.strip()
df["IngresoMin"] = df["IngresoMin"].apply(lambda x: x.replace("$",""))
df["IngresoMax"] = df["IngresoMax"].apply(lambda x: x.replace("$",""))
print(df)
```

Stage 7: Modeling

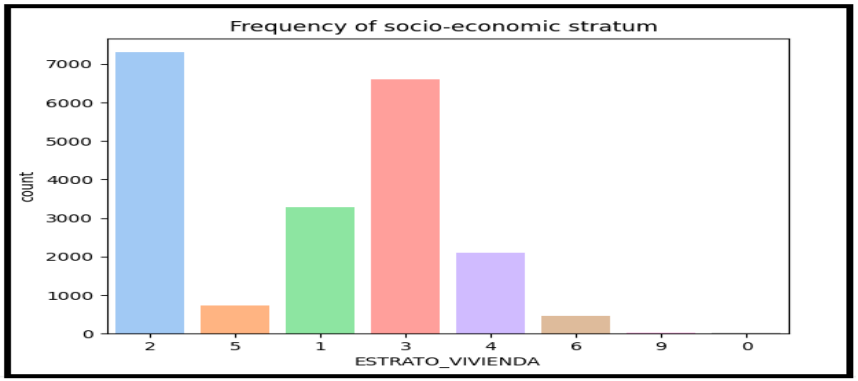


Stage 8: Evaluation and Implementation
Descriptive model

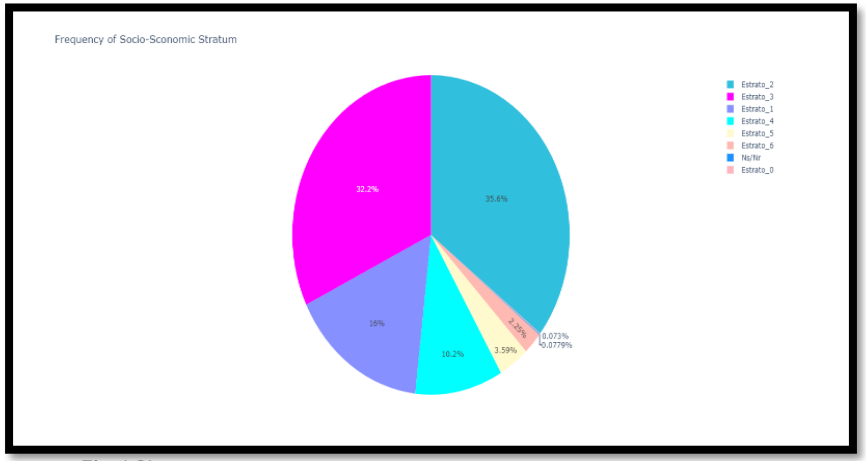
In this stage, in the first instance, all the procedures will be applied, such as diagnostic measures, graphs, statistical analysis and other processes leading to the implementation of the descriptive model; that allow to address the problem object of this work in an adequate and complete way, thus achieving the fulfillment of the objectives set.

Stage 8: Evaluation and Implementation
Descriptive model : Analysis of the socioeconomic stratum variable

It is evident that strata 2, 3, and 1 contain the highest percentage of population concentration, corresponding to the population with the lowest purchasing power and the lowest income (Table 1). Therefore, it can be inferred that in Colombia's main cities, the majority of the population is in a critical condition in socio-economic terms.

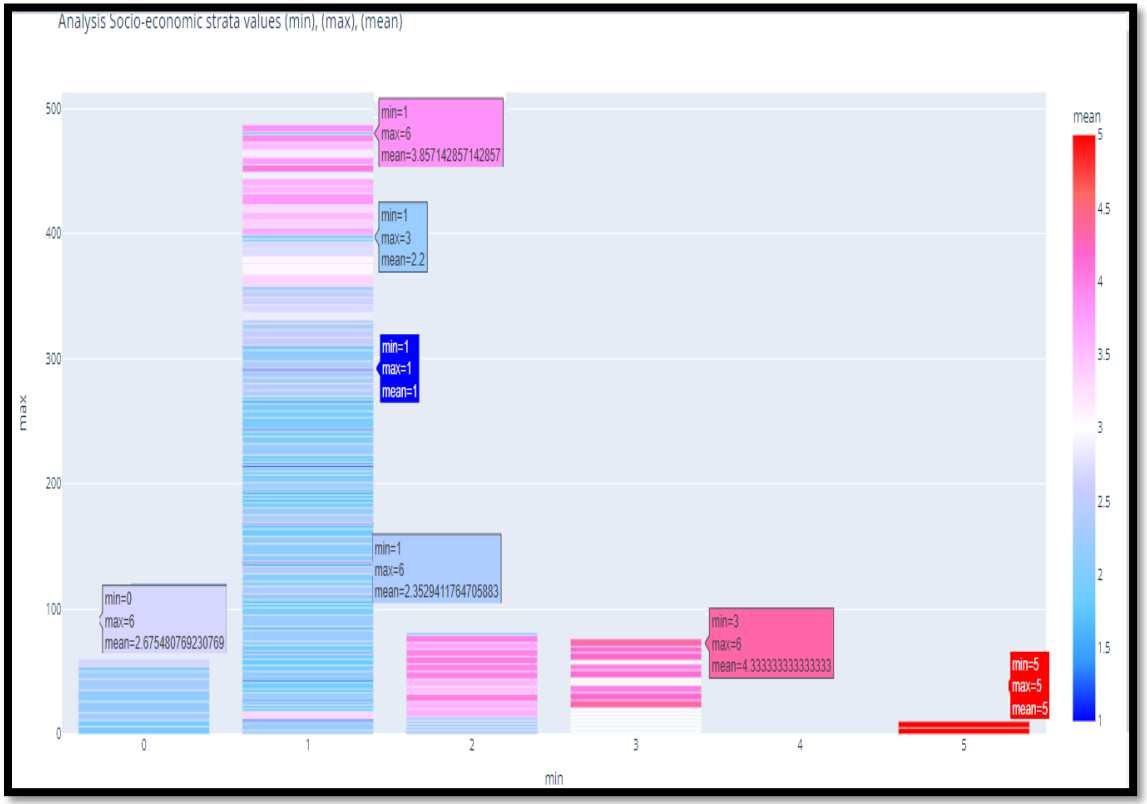


Stratum	Classification
1	Low-low
2	Low
3	Lower-middle
4	Medium
5	Medium-high
6	High



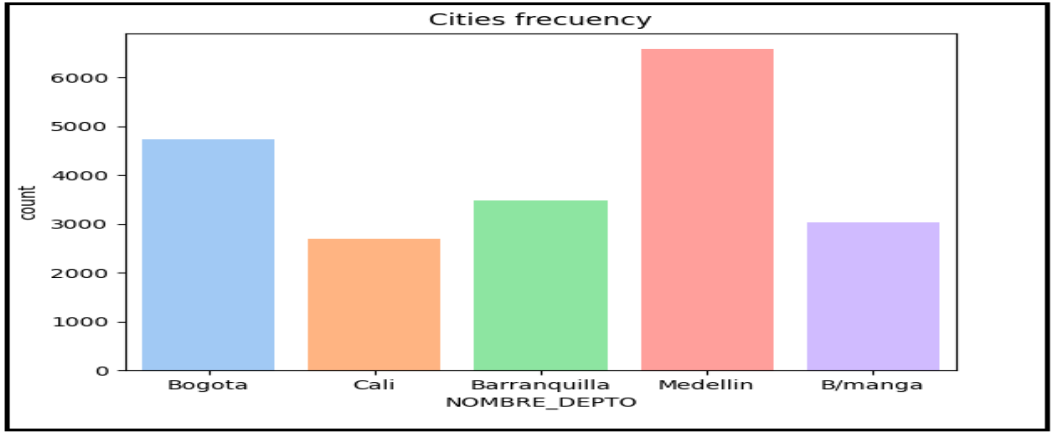
Stage 8: Evaluation and Implementation
Descriptive model : Analysis of the socioeconomic stratum variable

When calculating the means of the socio-economic stratum variable, the predominance of stratum 2 can be observed.

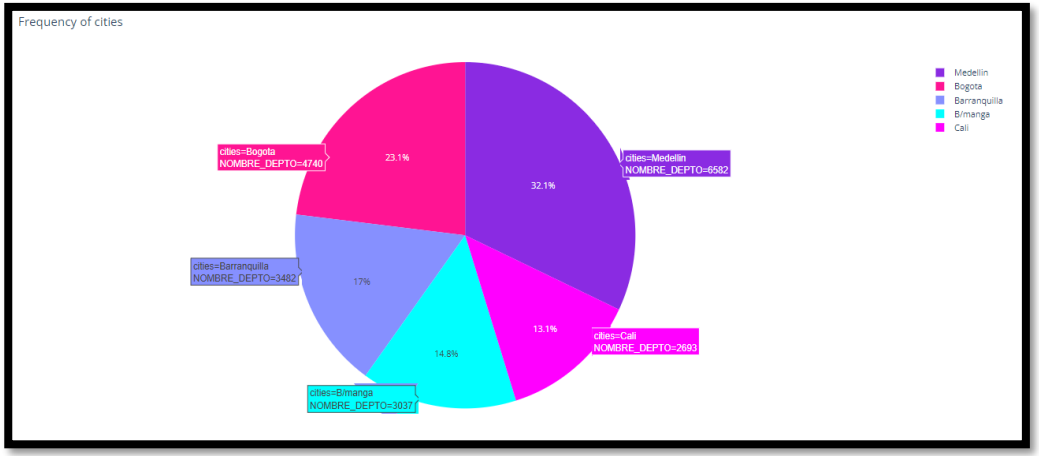


Stage 8: Evaluation and Implementation
Descriptive model : Analysis of the cities variable

It can be observed that the cities with the highest number of respondents are Medellin, Bogotá, and Barranquilla.



It can also be inferred that the number of respondents in the city of Medellin is double the number of respondents in the city of Cali



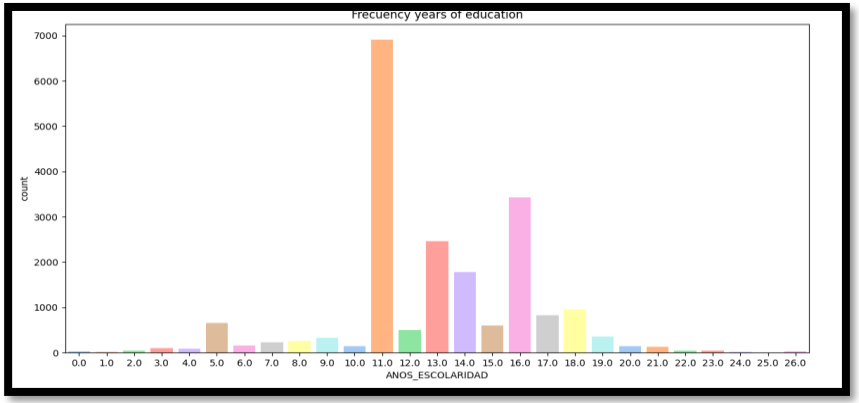
kaggle

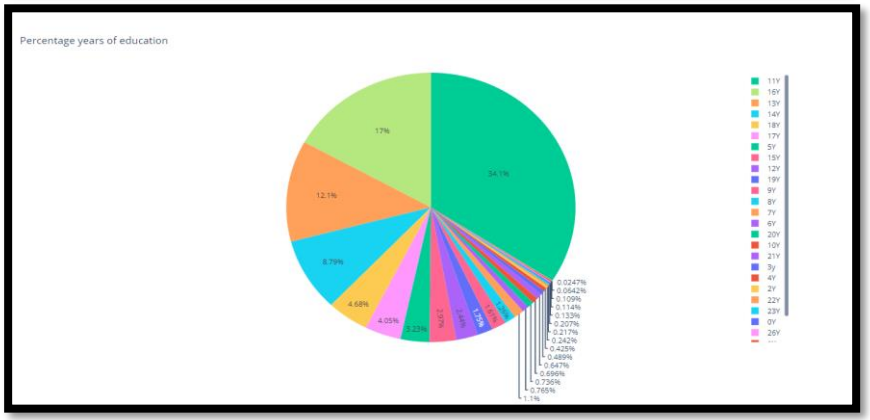
Stage 8: Evaluation and Implementation
Descriptive model : Analysis of the years of education variable

This variable corresponds to the years of formal academic education a citizen has completed. The variable takes values from 0 to 26

it can be seen that the highest frequency is at 11 years of schooling, which indicates that a large percentage of the surveyed population has a secondary education or less.

The following values with the highest frequencies correspond to 16, 13, and 14 years of age. This corresponds to technical and technological education.



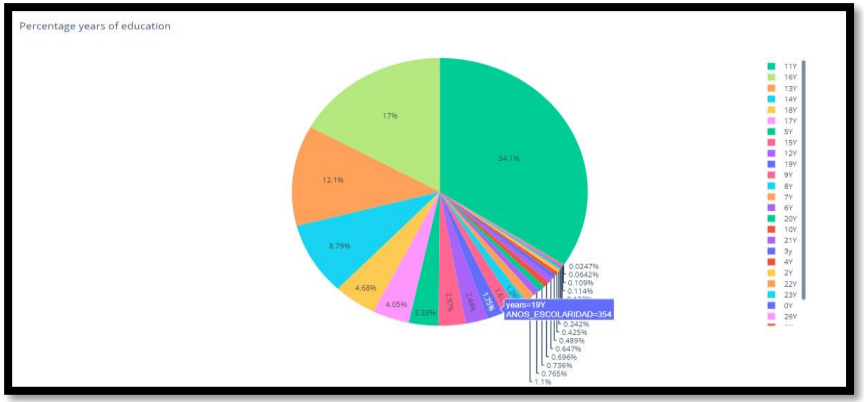


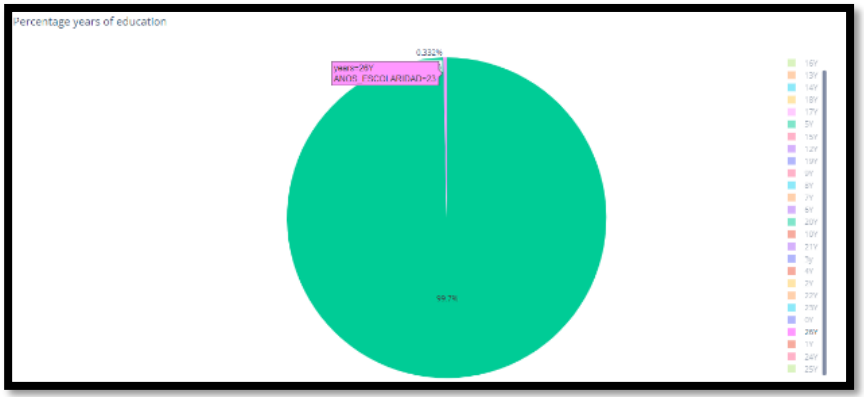
Stage 8: Evaluation and Implementation
Descriptive model : Analysis of the years of education variable

Only 1.75% of the population has 19 years of academic training corresponding to professional training.

It also shows that less than 0.1% of the population has postgraduate studies.

Finally, only 0.332% of the population has doctoral studies.





Stage 8: Evaluation and Implementation

Descriptive model : Analysis of the variable work hours per weekly

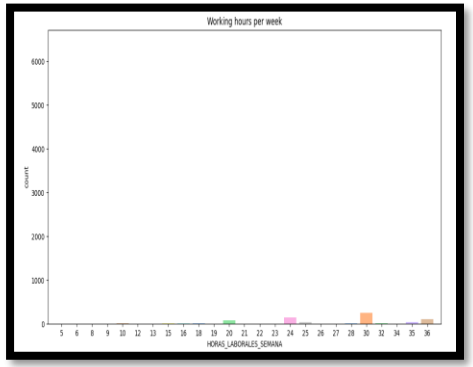
Analyzing this variable by ranges we have:

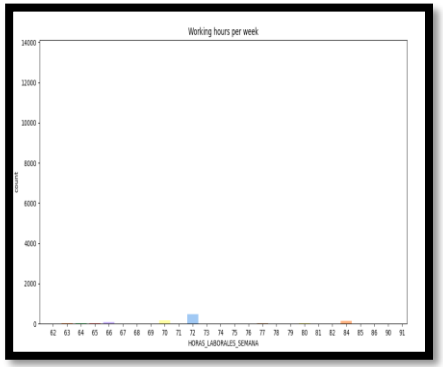
Range1. (0 - 36) Working hours per week

Range2. (37 - 61) Working hours per week

Range3. (62 - 91) Working hours per week

The Range2 shows the highest distribution of the population with respect to working hours per week. It can be seen that the value of 48 hours presents the highest frequency of occurrence, followed by 40, 60, 50, 45, 45, 56, 54, 55, 44, 42, and 52 hours as the highest frequency values.

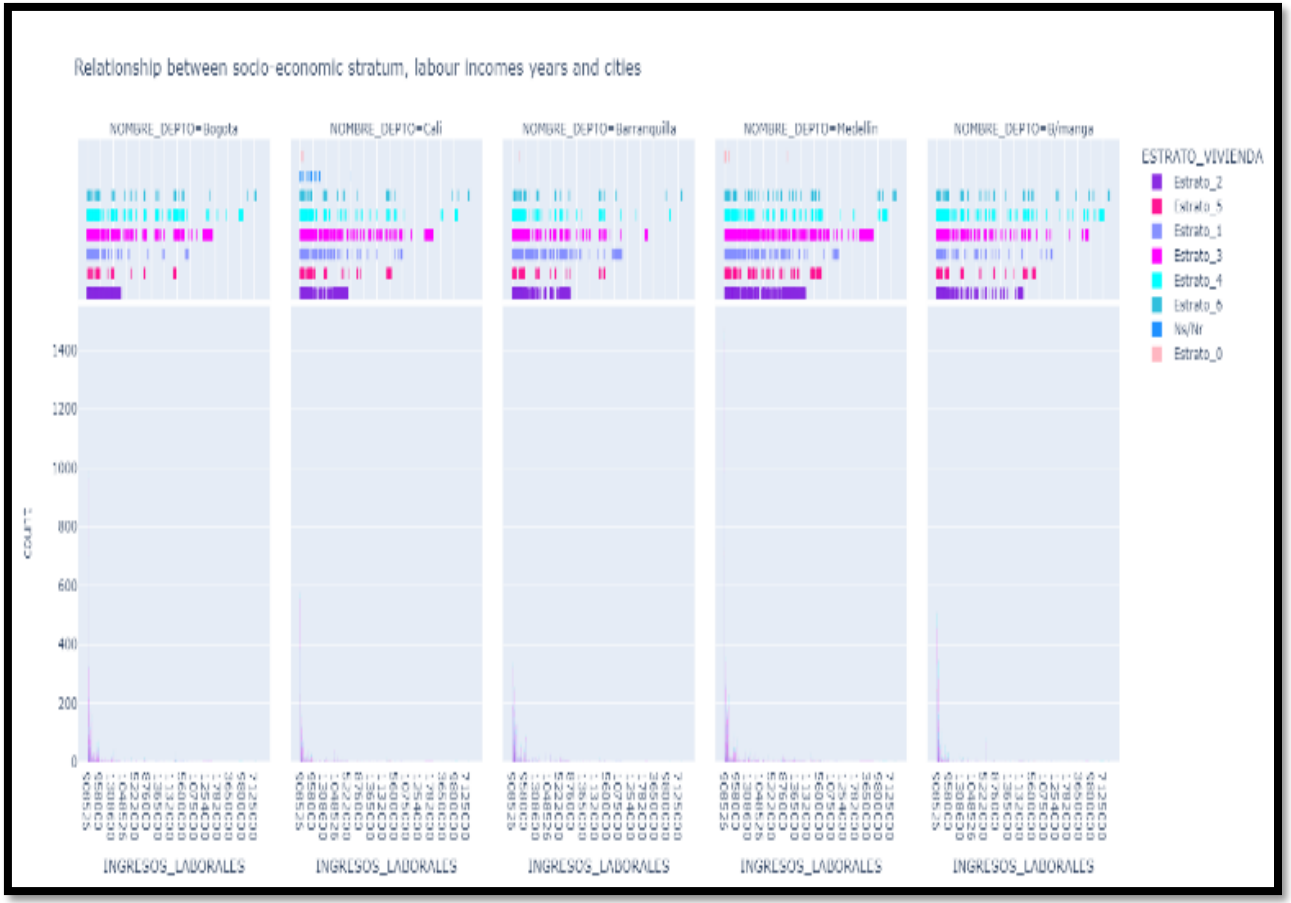




Stage 8: Evaluation and Implementation

Descriptive model :Analysis of the variable labor income

The following graph allows the analysis of the citizen's income with respect to the socio-economic stratum and the city in which the citizen resides.



kaggle

Stage 8: Evaluation and Implementation

Descriptive model :Analysis of the variable labor income

- **Next, emphasis is placed on the values of the study variables, as they present several multiple inconsistencies with respect to labor income and the socio-economic stratum where the citizen resides.**
- **It is possible to appreciate values that do not make sense even within the same city, some of the appreciations are described below.**
- **For example, in the city of Cali, the citizen who answered the survey belongs to stratum 6, i.e. the stratum with the highest purchasing power in Colombia, his household is made up of only himself and his income is $4300000. As of today's date, 18 March 2023, the dollar equivalent of the Colombian peso is 1 USD = 4848.12 COP, i.e. $4300000 corresponds to approximately 902.55 USD.**
- **In the same city, a second household surveyed has an income of $18000000 and is made up of two people, therefore, each member of the family has an average income of $9000000, which is equivalent to approximately 1889.06 USD; this family is located in stratum 5.**
- **This does not make sense since the second household has a higher income than the first citizen, in fact, it is more than double.**

Stage 8: Evaluation and Implementation

Descriptive model :Analysis of the variable labor income

- **Continuing with Cali, there is a household with an income of $950000 or 199.40 USD approximately, equivalent on average to one legal monthly minimum wage, regardless of the number of inhabitants of this household, and for this reason, it should belong to a stratum 1.**
- **The same phenomenon occurs in the city of Bogotá. There is a household in stratum 4 made up of 4 people and with an income of $1180000, with 4 inhabitants, the average is $295000, that is to say, 61.92 dollars on average.**
- **Continuing with Medellín, there is a household made up of a single citizen with an average monthly income of $4200000, or 881.56 USD approximately. This is more than 4 minimum wages and is stratified in level 2.**
- **The above examples are not isolated cases, however, in order to have more tools for judgment we proceed to analyze the variables.**

Stage 8: Evaluation and Implementation

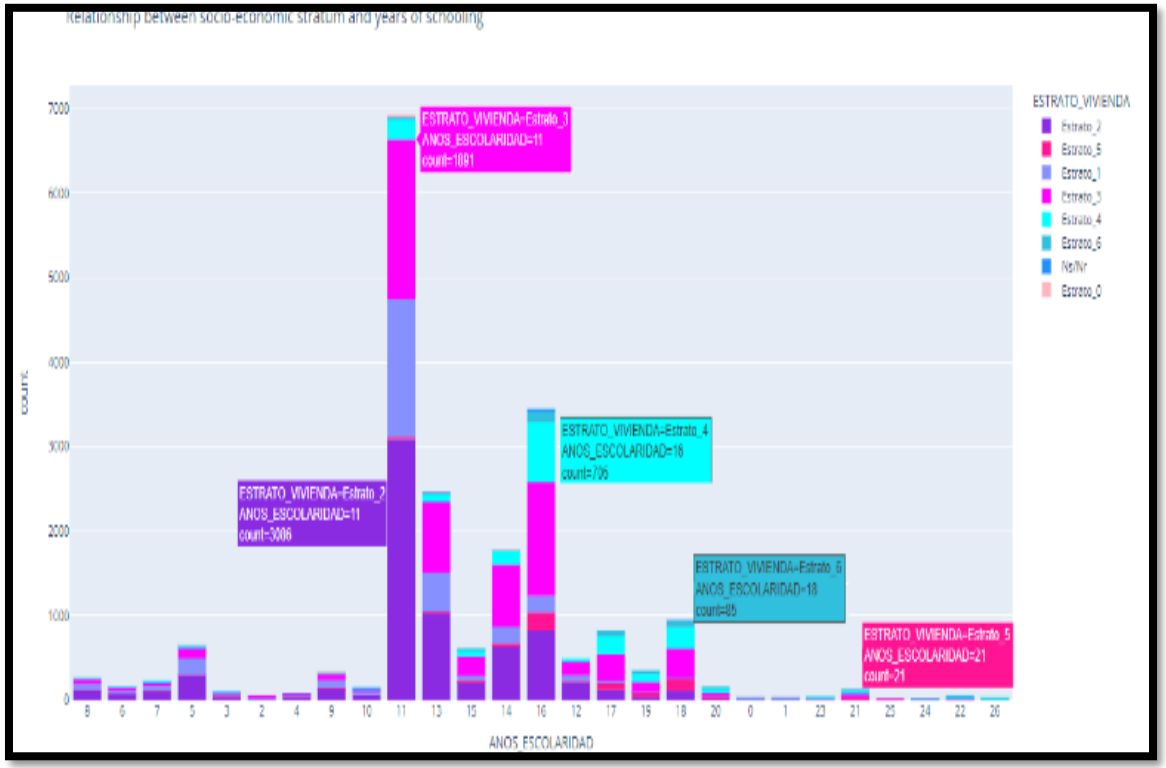
Descriptive model Relationship between the variable Labor income, socio-economic stratum and city with indicator values.

Stage 8: Evaluation and Implementation

Descriptive model : Relationship between years of schooling and socio-economic stratum

This graph shows that 11 years of schooling is the most frequently repeated frequency. It can also be seen that the strata that make up this category are 2, 3, and 1 in order of participation.

The low frequency of citizens with university and postgraduate studies is also observed. This confirms the analysis carried out previously.

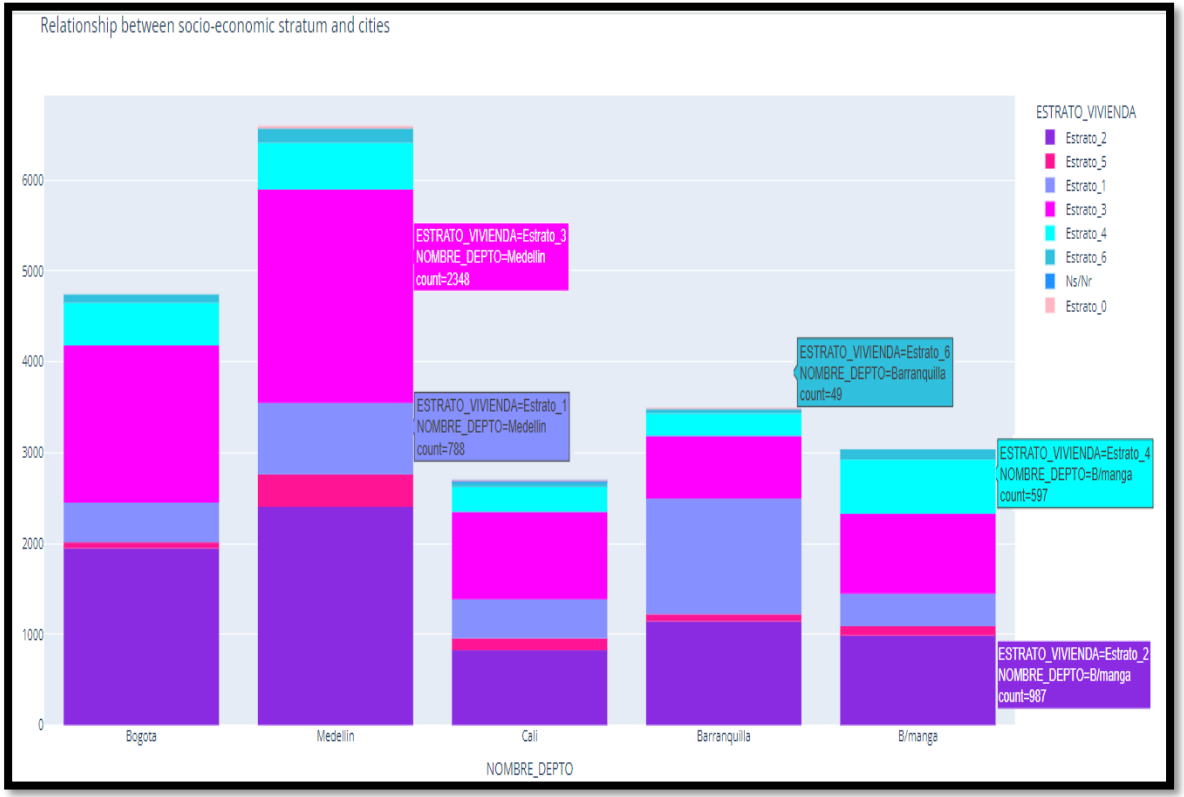


Stage 8: Evaluation and Implementation

Descriptive model : Relationship between years of schooling and socio-economic stratum

The graph shows the relationship between the socio-economic stratum and the cities where the GEIH was implemented.

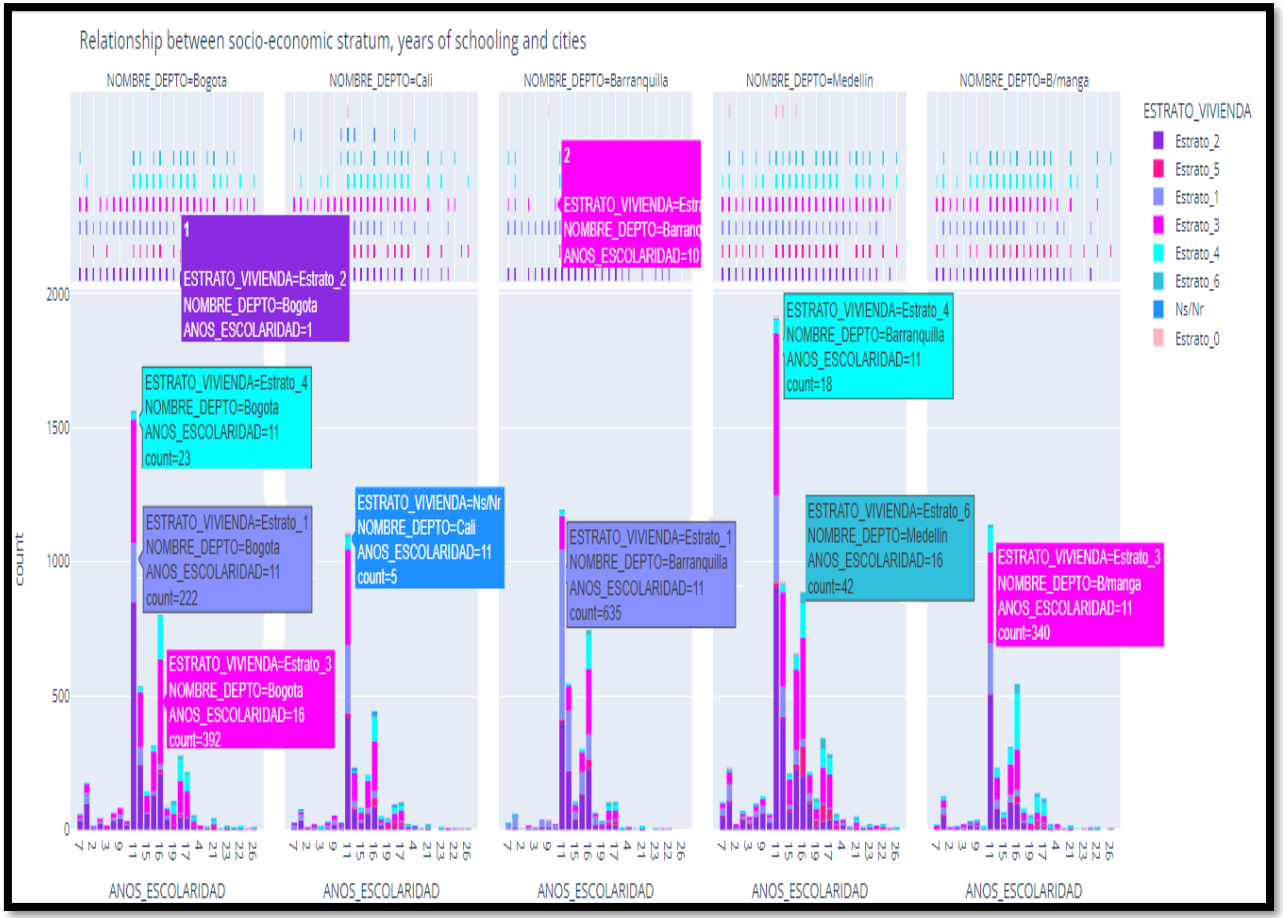
It can be seen that the highest frequency is in the city of Medellín, followed by Bogotá, Barranquilla, Bucaramanga, and finally Cali.



Stage 8: Evaluation and Implementation

Descriptive model : Relationship between years of schooling and socio-economic stratum

This graph clearly shows that strata 1,2,3 have predominant participation in all the cities, in contrast to strata 4,5, and 6 with higher incomes.



kaggle

Stage 8: Evaluation and Implementation

Descriptive model : Statistical analysis of the average income variable

The average income variable is created, which is the result of the division between the variables of labor income and the number of persons per household. in this way, the income for each person in the household is obtained.

Next, the statistical analysis of the average income variable is implemented with respect to the age of the surveyed citizen and the socio-economic stratum in which he/she was classified.

```
df2 = pd.read_csv('GEIH.csv', index_col=0)

df2['Ingreso_Promedio'] = df2['INGRESOS_LABORALES'] / df2['N_PERSONAS_HOGAR']
print(df2.Ingreso_Promedio)
df2Estrato_Vivienda1 =df2.groupby(['EDAD_ANOS','ESTRATO_VIVIENDA' ]).Ingreso_Promedio.agg([len, max, min, np.mean, np.std])
pd.options.display.max_rows = None
print(df2Estrato_Vivienda1)
```

EDAD_ANOS	ESTRATO_VIVIENDA	len	max	min	mean	std
18.0	1	14	5.000000e+05	5.333333e+04	2.105034e+05	1.334747e+05
	2	36	4.600000e+05	6.500000e+04	2.287504e+05	1.193980e+05
	3	22	1.200000e+06	9.600000e+04	2.991894e+05	2.288247e+05
	4	1	1.817052e+05	1.817052e+05	1.817052e+05	NaN
19.0	0	1	5.250000e+05	5.250000e+05	5.250000e+05	NaN
	1	41	4.542630e+05	2.600000e+04	2.088392e+05	8.494200e+04
	2	67	9.084540e+05	3.333333e+04	2.781274e+05	1.627910e+05
	3	57	2.500000e+06	8.259327e+04	3.816962e+05	3.767657e+05
	4	3	3.028420e+05	1.066667e+05	1.731696e+05	1.123120e+05
	5	1	2.416667e+05	2.416667e+05	2.416667e+05	NaN
20.0	1	64	1.700000e+06	6.363636e+04	2.995709e+05	2.783701e+05
	2	110	1.700000e+06	9.085260e+04	3.368742e+05	2.531054e+05
	3	92	1.250000e+06	3.950113e+04	3.275568e+05	2.073703e+05
	4	16	5.500000e+05	1.514210e+05	2.686014e+05	1.039970e+05
	5	3	6.000000e+05	2.450000e+05	3.816667e+05	1.910715e+05
	9	1	2.271315e+05	2.271315e+05	2.271315e+05	NaN
21.0	0	1	1.817052e+05	1.817052e+05	1.817052e+05	NaN
	1	81	1.300000e+06	3.000000e+04	3.239314e+05	2.551504e+05
	2	151	1.600000e+06	7.500000e+04	3.428424e+05	2.232374e+05
	3	96	9.100000e+06	6.666667e+04	4.351217e+05	1.028821e+06
	4	25	1.150000e+06	1.817052e+05	4.308701e+05	2.624945e+05
	5	5	1.050000e+06	1.510000e+05	4.567631e+05	3.751933e+05
	6	1	8.950000e+05	8.950000e+05	8.950000e+05	NaN
22.0	0	1	4.542630e+05	4.542630e+05	4.542630e+05	NaN
	1	103	1.000000e+06	3.600000e+04	2.970989e+05	1.755874e+05
	2	198	2.500000e+06	7.538462e+04	3.653330e+05	2.995191e+05
	3	154	3.000000e+06	7.500000e+04	4.214214e+05	3.658021e+05
	4	30	9.085260e+05	8.333333e+04	3.553157e+05	1.724277e+05
	5	4	3.400000e+05	2.271315e+05	3.018684e+05	5.263137e+04
	6	5	2.900000e+06	2.275000e+05	9.819000e+05	1.112421e+06
	9	1	2.271315e+05	2.271315e+05	2.271315e+05	NaN

Stage 8: Evaluation and Implementation

Descriptive model : Statistical analysis of the average income variable

According to the survey on the characterization of monetary poverty and results of social classes in 2020. DANE defines social classes according to the income generated by a household during a month, this income is divided by the number of inhabitants that make up the household. From this point, the following categories are determined:

- Poor class: if by dividing the total income by the number of household members the figure is equal to or less than $331,688 pesos per person (82.23 USD), those citizens are considered to be part of the poor class.

- Poor class: if, when dividing the total income by the number of household members, the figure is equal to or less than $331,688 pesos per person (82.23 USD), these citizens are considered to be part of the poor class.

- Vulnerable class: if, when dividing the total income by the number of household members, the figure is in the range of $331,688 (82.23 USD) to $653,781 (162.09 USD) pesos per person, these citizens are considered to be part of the vulnerable class.

- Middle class: if, when dividing the total income by the number of household members, the figure is in the range of $653,781 (162.09 USD) to $3'520,360 (872.43 USD) pesos per person, these citizens are considered to be part of the middle class.

- Upper class: If the total income divided by the number of household members is more than $3'520,360 (872.43 USD) pesos per person, these citizens are considered to be in the upper class.

Stage 8: Evaluation and Implementation

Descriptive model : Statistical analysis of the average income variable

- **The first thing we can analyze throughout the entire output is that the average income per person in the household (min) is mostly low.**
- **For the whole output range (18 - 63) years, the values of average income per person per month (min) are mostly very low, even below $100000 (22.56 USD). This is even below the poverty range provided by DANE. This behavior is present indistinctly between age ranges.**
- **Throughout the whole set of outputs, a disturbing behavior is observed; the values of average monthly income per person (min) do not vary proportionally to the socio-economic stratum in which they are classified. One would expect the minimum monthly income (min) to increase as the level of stratification increases, but this does not happen.**
- **There is a very marked difference between the values of the average monthly income per person (min) and the average monthly income per person (max) for the same stratum, even more than ten times. This fact is repeated regardless of the age of the citizen and is more critical in the lower socio-economic strata 1, 2, and 3.**

Stage 8: Evaluation and Implementation

Descriptive model : Statistical analysis of the average income variable

- **Another critical situation is observed with respect to the values of average monthly income per person (max). It is observed that in the lower strata, 1, 2, and 3 these values exceed the social class ranges established by DANE. Thus, citizens who have incomes that place them in the middle and upper classes belong to the lower socio-economic stratification that should correspond to the poor and vulnerable population.**
- **This phenomenon is observed throughout the whole of the exit, not in isolation.**

- **In this way, the subsidies and social programs delivered to socio-economic strata 1, 2, and 3 are reaching the middle and upper social classes and not the vulnerable population for whom they should be intended.**
- **Another relevant phenomenon is that which occurred in strata 5 and 6 with respect to the values of average income per person per month (min) and average income per person per month (max). It is observed that these values are classified within the social classes; vulnerable and even poor.**

Stage 8: Evaluation and Implementation

Descriptive model : Statistical analysis of the average income variable

We can observe in the following figure how in stratum 6 average incomes corresponding to the vulnerable social class and even to the poor class are recorded.

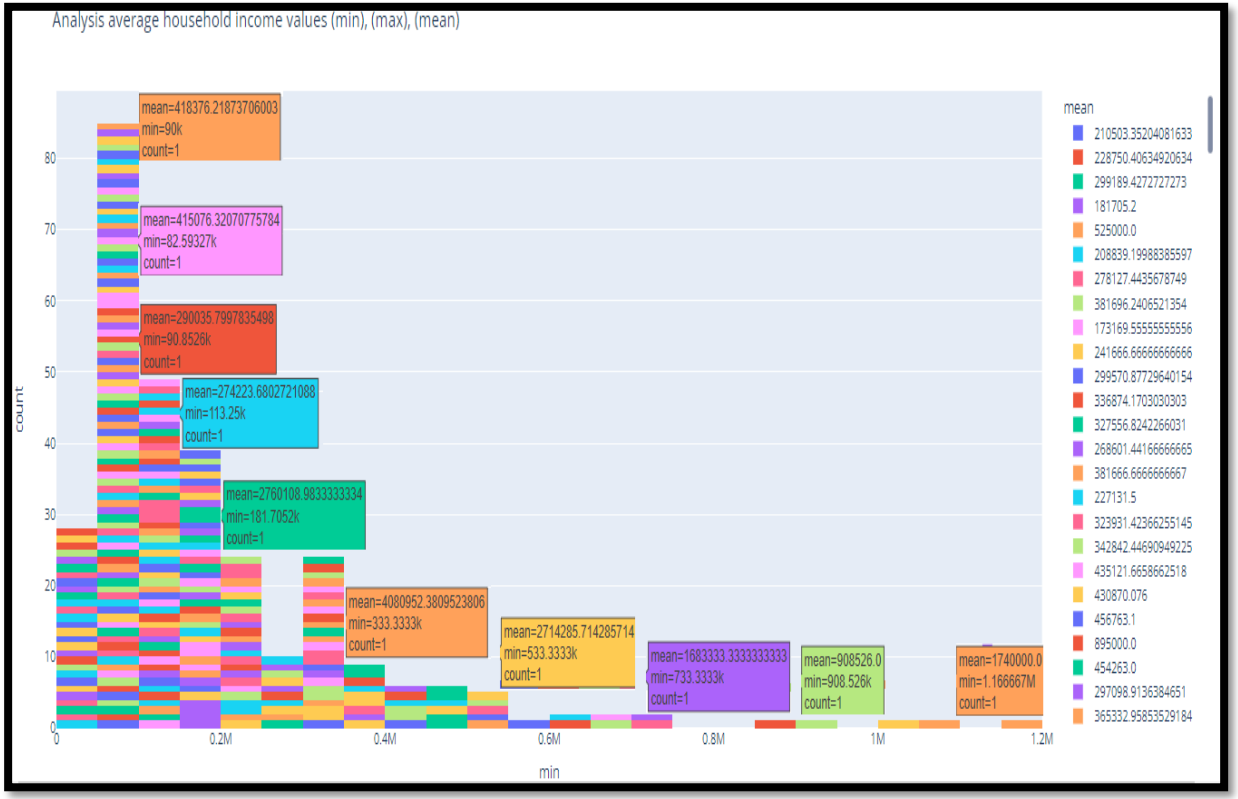
In all the labels it is evident that the average income recorded that classifies citizens socially does not correspond to the socio-economic stratification to which they belong.

Stage 8: Evaluation and Implementation

Descriptive model : Statistical analysis of the average income variable

With regard to the mean, it can be observed that its value is generally lower in the lower strata 1, 2, and 3 than the value presented in the higher strata 4, 5, and 6.

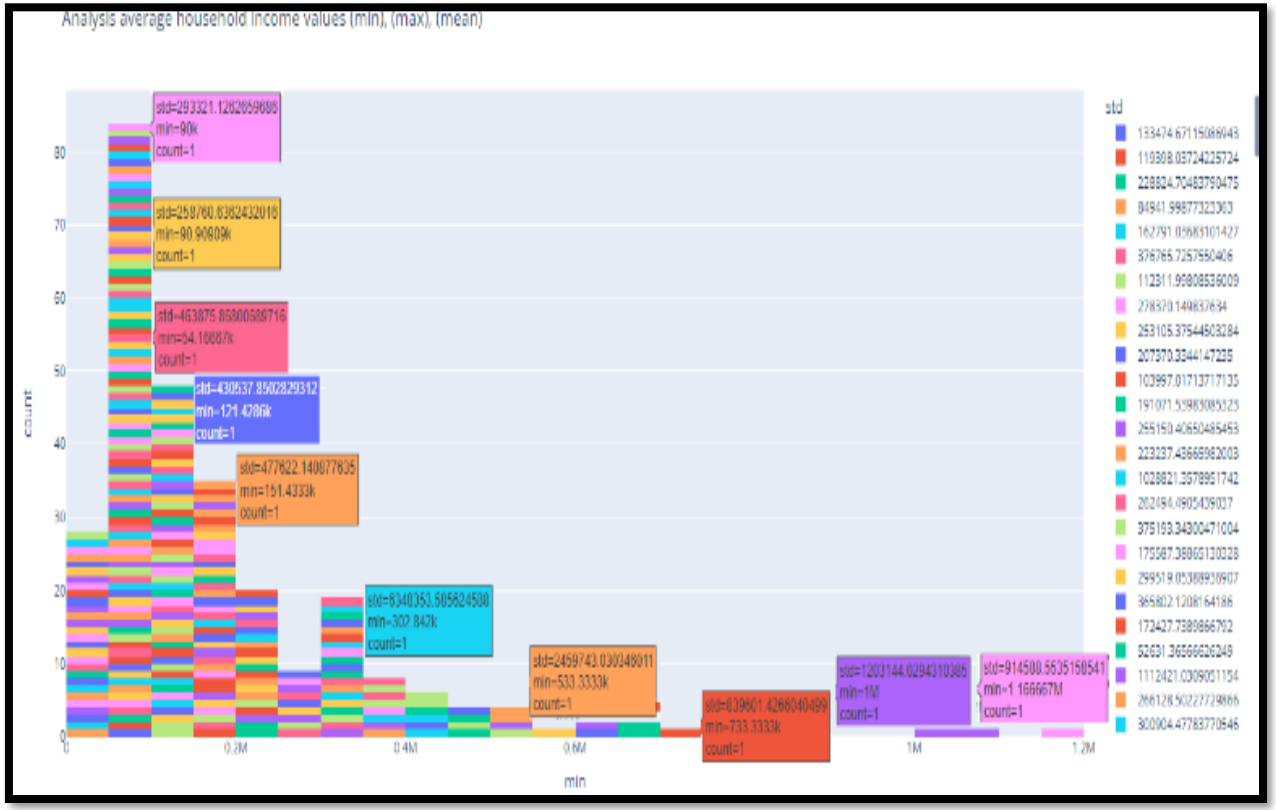
The vulnerable population should have low incomes and the middle and upper-class populations should have high incomes.



Stage 8: Evaluation and Implementation

Descriptive model : Statistical analysis of the average income variable

Finally, it can be observed that the standard deviation values present high values, which allows us to conclude that the distribution of the data is dispersed with respect to its mean.



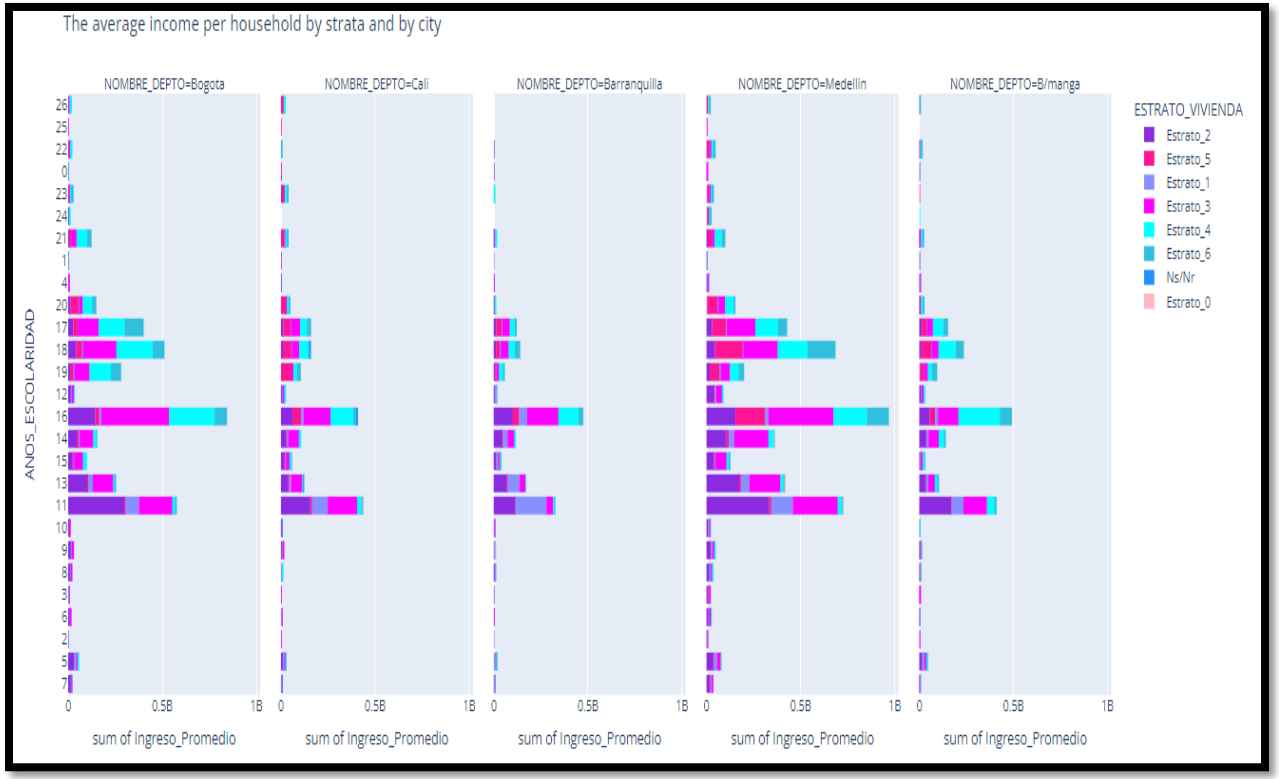
Stage 8: Evaluation and Implementation

Descriptive model : Relationship between average income and years of schooling by city.

The city of Medellín has the highest average income distribution, followed by Bogotá, Bucaramanga, Barranquilla, and finally Cali.

The presence of low socio-economic strata 1, 2, and 3 are predominant and corroborates the analysis carried out in the variable Name Department.

Likewise, the low population density is evident for the high values of the variable Years of Schooling. Values such as 11, 13, 16, 17, and 18 years show the highest concentrations.

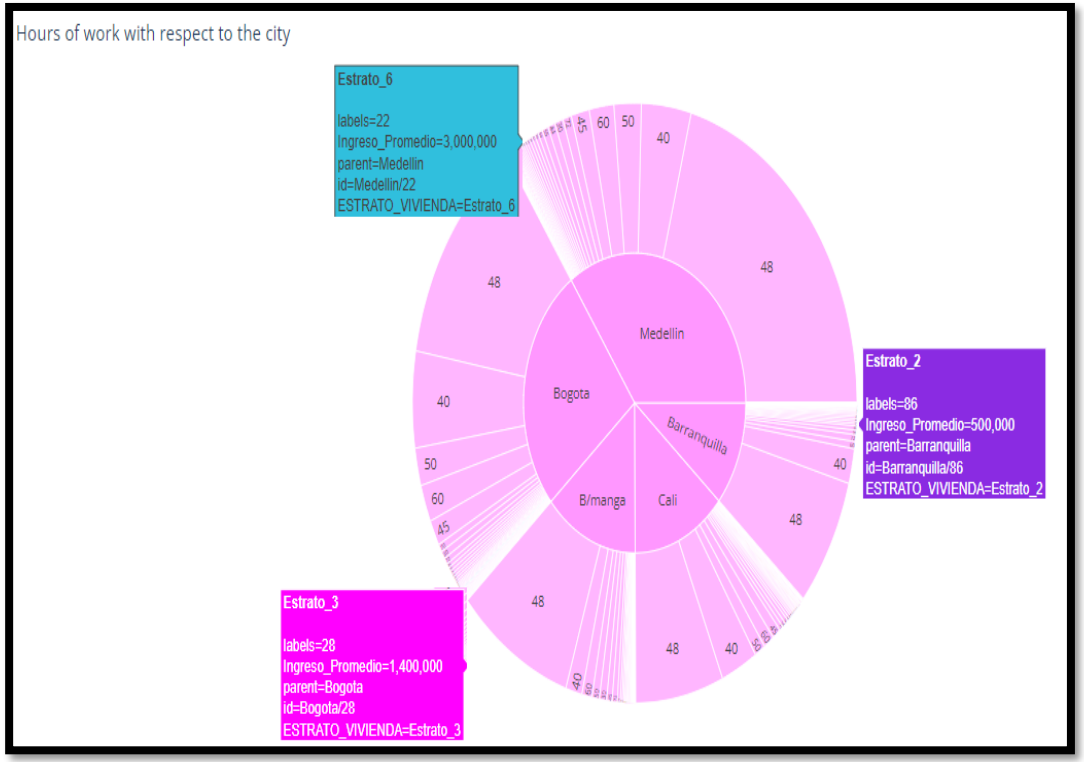


Stage 8: Evaluation and Implementation

Descriptive model : Relationship between cities and working hours

Regarding the relationship between cities and working hours. As can be seen in the following graph, for all cities 48 working hours per week is the value with the highest percentage of representation.

This is an expected value since Colombia establishes 48 maximum working hours for formal employment.



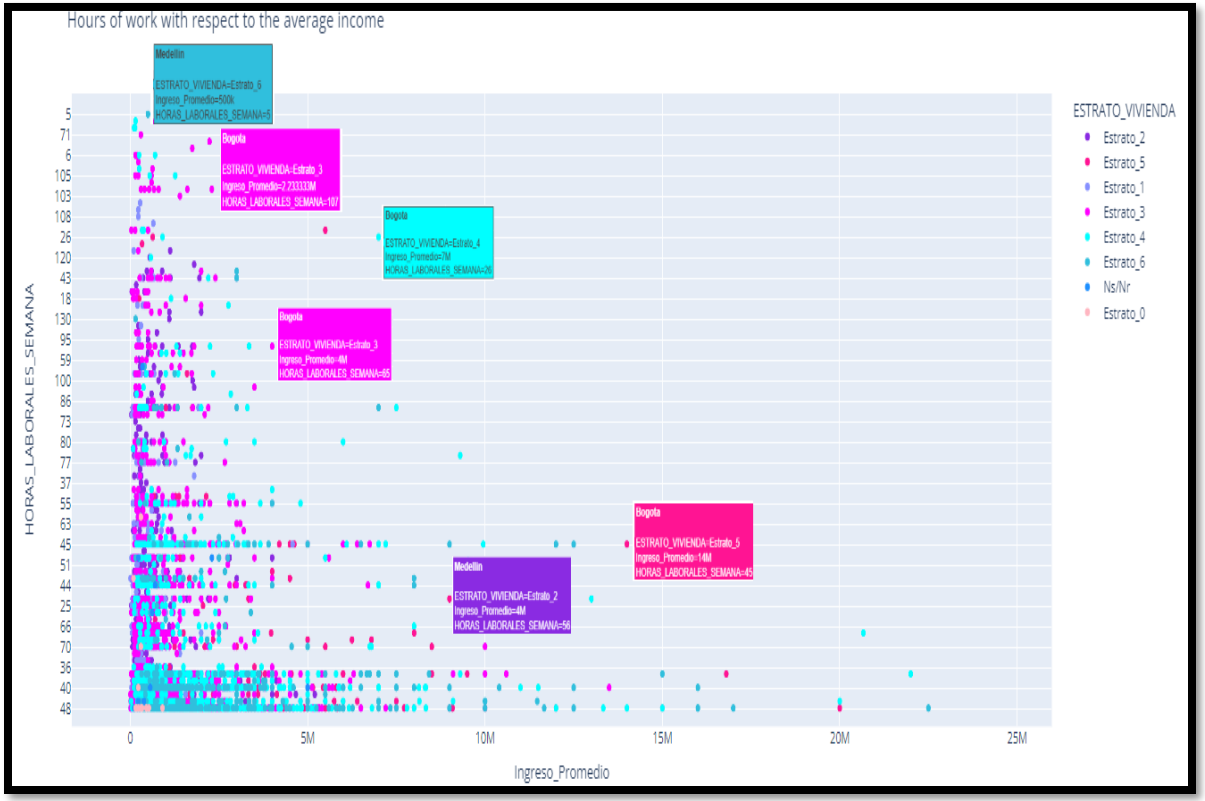
Stage 8: Evaluation and Implementation

Descriptive model: Relation between hours worked per week and average income

The following graph shows more clearly the distribution of average income vs. hours worked per week, broken down by stratum and city.

There is a high concentration of the population in the range (40-48) working hours.

There is also a notable presence of workers with an hourly workload between (60-70) hours. There are also hourly workloads above 70 hours, which concentrate a large proportion of the city.



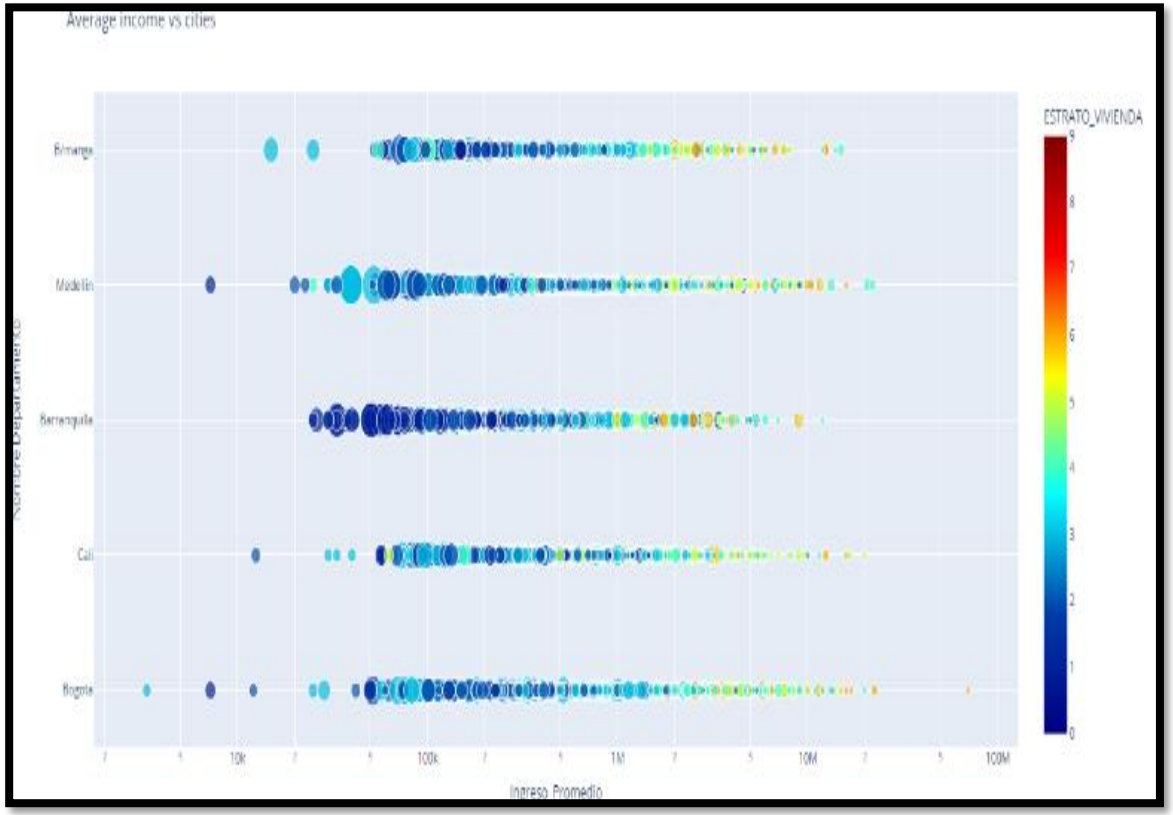
Stage 8: Evaluation and Implementation

Descriptive model: Relation between hours worked per week and average income

The following scatter graph provides an in-depth analysis of the behavior of the average income in relation to the city.

In the following graph, it can be seen that the highest population density in the dataset belongs to the lower strata 1, 2, and 3.

We proceed to analyze the sectors with the lowest and highest average income.



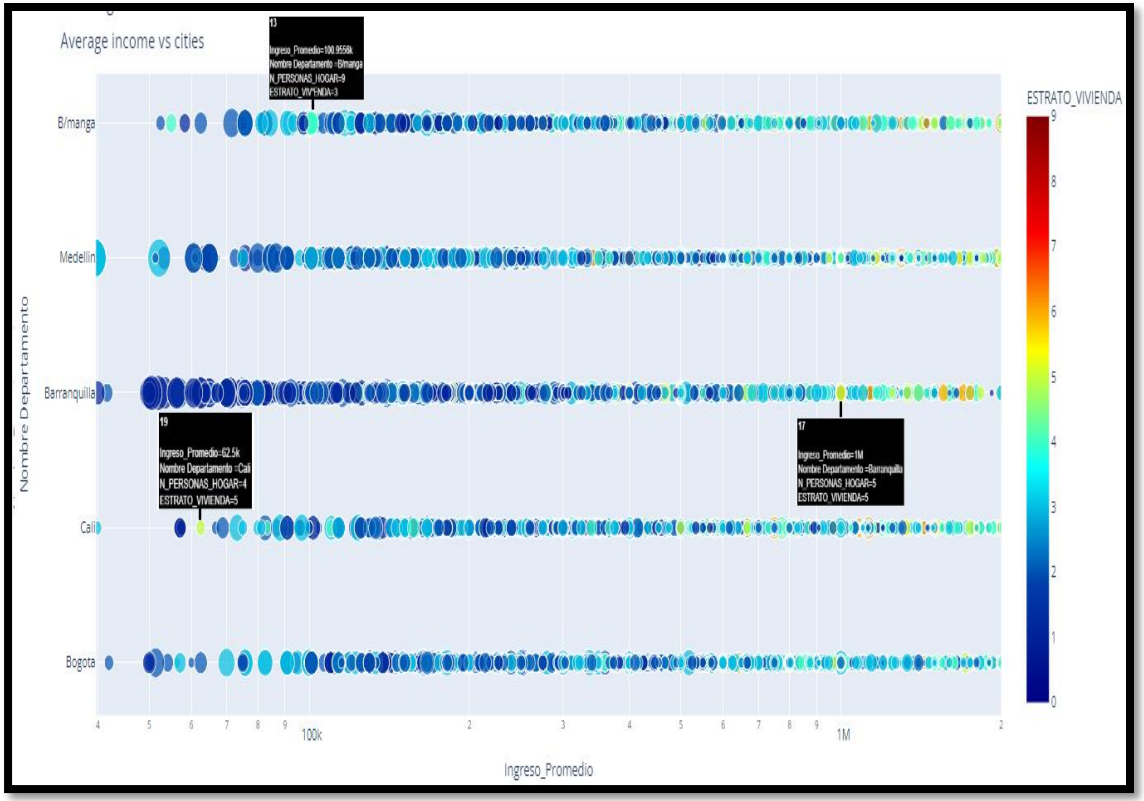
Stage 8: Evaluation and Implementation

Descriptive model: Relationship between average income up to 2 million pesos (USD 445) and cities

The following graph specifies average incomes up to two million pesos (445.87 USD).

The predominance of strata 1 and 2 in the selected distribution is observed, however, once again the labels alert us to situations that indicate an error in the targeting method. Thus, a household in Barranquilla composed of 3 people is stratified as 5, but its average income is 1 million pesos per person (247.93 USD).

It is also observed that a household in Cali composed of 4 persons is stratified in grade 5, but its average income per person is 62500 pesos (15.49 USD).



Stage 8: Evaluation and Implementation

Descriptive model: Ratio of average income between 2 million pesos (445.87 USD) and 5 million pesos (1239.69 USD) by city.

We proceed to analyze the range between 2 million pesos (445.87 USD) and 5 million pesos (1239.69 USD).

In this range of analysis, the inconsistencies between socio-economic stratification and average income are more than evident. The labels allow us to be certain about the analysis.

Thus a household in Bogotá consisting of 4 persons, with an average income per person of 2.5 million pesos (619.84 USD) is socio-economically stratified in level 2.

A second household in Medellín consisting of 2 persons, with an average income per person of 3.5 million pesos (867.78 USD) is socio-economically stratified in level 3.

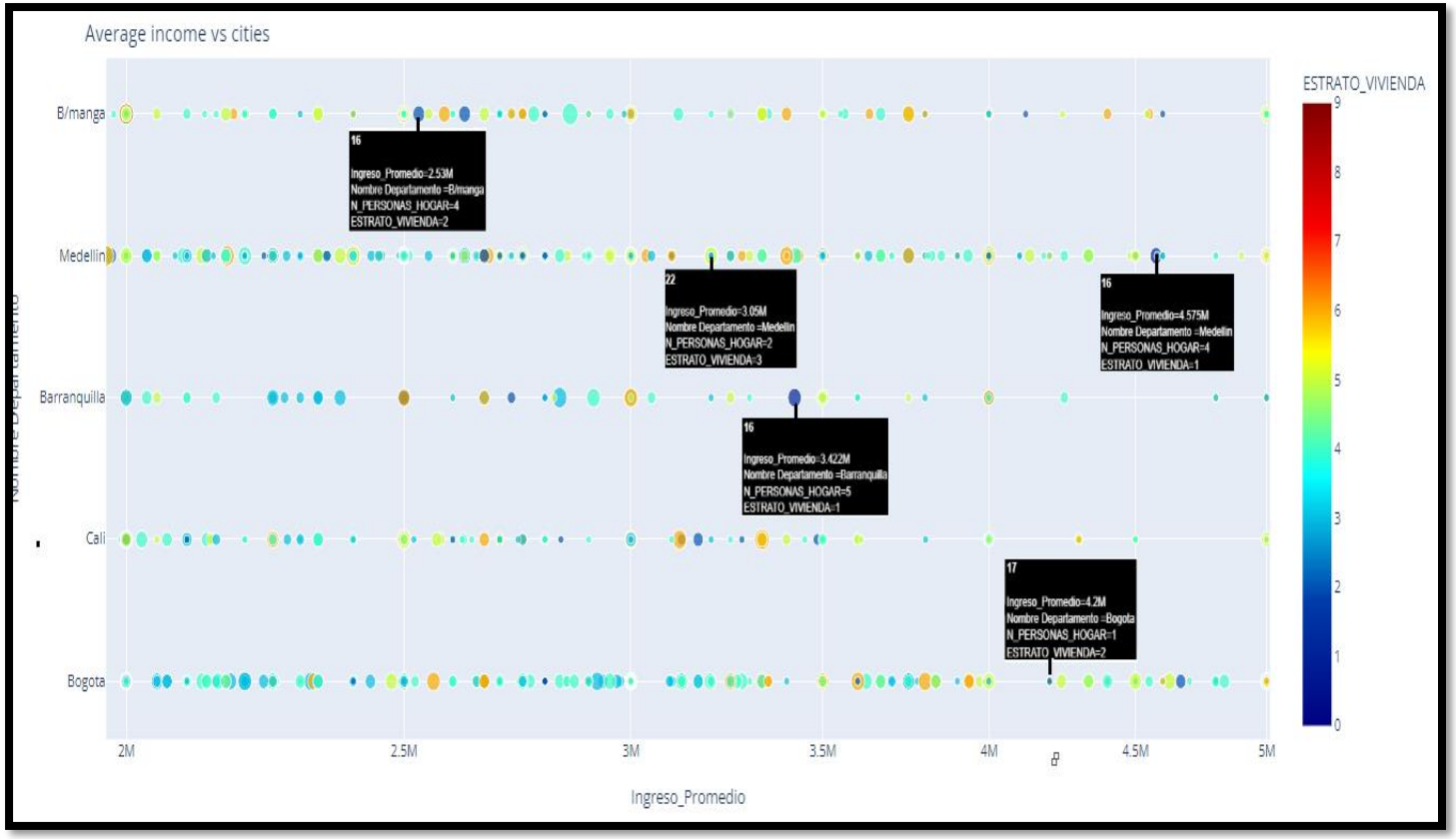
A third household in Medellín consisting of 4 persons, with an average income per person of 4.57 million pesos (1133.08 USD) is socio-economically stratified in level 1.

A fourth household in Barranquilla consisting of 5 persons, with an average income per person of 3.42 million pesos (847.95 USD) is socio-economically stratified in level 1.

Stage 8: Evaluation and Implementation

Descriptive model: Ratio of average income between 2 million pesos (445.87 USD) and 5 million pesos (1239.69 USD) by city.

This behavior is again consistent across the selected range. This indicates that many citizens belonging to the middle and upper social classes have socio-economic strata that should correspond to the poor and vulnerable population.



Stage 8: Evaluation and Implementation
Descriptive model: Decision tree model

First, the DataFrame on which the predictions will be implemented is created

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode', index_col=0)

df2 [ "INGRESOS_LABORALES" ]  =  df2 [ "INGRESOS_LABORALES" ] . astype ( float )
df2 [ "N_PERSONAS_HOGAR" ]  =  df2 [ "N_PERSONAS_HOGAR" ] . astype ( float )
df2 [ "HORAS_LABORALES_SEMANA" ]  =  df2 [ "HORAS_LABORALES_SEMANA" ] . astype ( float )
df2 [ "EDAD_ANOS" ]  =  df2 [ "EDAD_ANOS" ] . astype ( float )
df2 [ "ANOS_ESCOLARIDAD" ]  =  df2 [ "ANOS_ESCOLARIDAD" ] . astype ( float )
df2 [ "N_HIJOS" ]  =  df2 [ "N_HIJOS" ] . astype ( float )
df2 [ "ESTRATO_VIVIENDA" ]  =  df2 [ "ESTRATO_VIVIENDA" ] . astype ( float )

df2["Ingreso_Promedio"] = df2["INGRESOS_LABORALES"] / df2["N_PERSONAS_HOGAR"]
df2.columns = df2.columns.str.strip()
df2["Ingreso_Promedio"] = pd.to_numeric(df2["Ingreso_Promedio"], downcast="float")

df2_Final = df2.loc[ : , ['INGRESOS_LABORALES', 'N_PERSONAS_HOGAR', 'HORAS_LABORALES_SEMANA', 'N_HIJOS',
                          'Ingreso_Promedio','ESTRATO_VIVIENDA'  ]]
print(df2_Final.info())
```

Stage 8: Evaluation and Implementation

Descriptive model: Decision tree model

Next, the prediction target is selected, denoted by the variable y. In this case, the prediction target selected is the variable ESTRATO_VIVIENDA.

In the same way in this stage, the characteristics are selected; these are columns that are introduced in the model which will be used to make predictions, they are denoted with the letter X.

```
y = df2_Final.ESTRATO_VIVIENDA

df2_features = ['INGRESOS_LABORALES', 'N_PERSONAS_HOGAR', 'HORAS_LABORALES_SEMANA', 'N_HIJOS','Ingreso_Promedio'  ]

X = df2_Final[df2_features]

print(X.describe())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\ml3.py
<class 'pandas.core.frame.DataFrame'>
Index: 20534 entries, 2021025368326-1-1 to 2021045428964-1-2
Data columns (total 6 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   INGRESOS_LABORALES     20534 non-null  float64
 1   N_PERSONAS_HOGAR       20534 non-null  float64
 2   HORAS_LABORALES_SEMANA 20534 non-null  float64
 3   N_HIJOS                20534 non-null  float64
 4   Ingreso_Promedio       20534 non-null  float32
 5   ESTRATO_VIVIENDA       20534 non-null  float64
dtypes: float32(1), float64(5)
memory usage: 1.0+ MB
```

Stage 8: Evaluation and Implementation

Descriptive model: Decision tree model

Next, the library scikit-learn is used for the implementation of this model.

We also define the type of model, in this case, it is a decision tree.

We proceed to make an initial prediction:

```
from sklearn.tree import DecisionTreeRegressor

df2_Final_model = DecisionTreeRegressor(random_state=1)
print(df2_Final_model.fit(X, y))

print(df2_Final_model.predict(X))
```

	INGRESOS_LABORALES	N_PERSONAS_HOGAR	HORAS_LABORALES_SEMANA	N_HIJOS	Ingreso_Promedio
count	2.053400e+04	20534.000000	20534.000000	20534.000000	2.053400e+04
mean	1.871867e+06	3.695432	48.774764	1.316499	6.997435e+05
std	2.168388e+06	1.819568	8.643610	1.042002	1.165251e+06
min	9.999000e+03	1.000000	4.000000	0.000000	3.333000e+03
25%	9.085260e+05	3.000000	48.000000	1.000000	2.285714e+05
50%	1.200000e+06	3.000000	48.000000	1.000000	3.750000e+05
75%	2.000000e+06	4.000000	48.000000	2.000000	7.095833e+05
max	7.000000e+07	23.000000	130.000000	9.000000	7.000000e+07

DecisionTreeRegressor(random_state=1)

Stage 8: Evaluation and Implementation

Descriptive model: Decision tree model - Model Prediction

The next step is to evaluate the constructed model. In most applications, the relevant measure of model quality is predictive accuracy.

With the MAE metric, the absolute value of each error is taken. This converts each error into a positive number. The average of these absolute errors is then taken. This will be the quality measure of the model

Once the model has been created, the mean absolute error is calculated.

```
from sklearn.metrics import mean_absolute_error
predicted_Stratum_values = df2_Final_model.predict(X)
print(mean_absolute_error(y, predicted_Stratum_values))
```

```
0.4853747123436424
```

Stage 8: Evaluation and Implementation

Descriptive model: Decision tree model - Model Prediction

Based on the measurement obtained, the data is divided into two parts. The data in the first group will be used as training data to fit the model. The data from the second group will be used as validation data to calculate mean_absolute_error.

The data obtained shows a significant improvement. Of course, this will generate a more accurate prediction.

A comparison of the two outputs is presented, the difference is noticeable.

```
from sklearn.model_selection import train_test_split
train_X, val_X, train_y, val_y = train_test_split(X, y, random_state = 0)
# Define model
df2_Final_model = DecisionTreeRegressor()
# Fit model
df2_Final_model.fit(train_X, train_y)
# get predicted prices on validation data
val_predictions = df2_Final_model.predict(val_X)
print(mean_absolute_error(val_y, val_predictions))
```

```
0.8529643668585362
```

```
DecisionTreeRegressor(random_state=1)
[2.17449664 5.         2.53333333 ... 1.95652174 2.         2.86956522]
0.485374712343642 4
[2.18061674 5.         2.41304348 ... 2.21428571 2.         2.9        ]
0.8529722508068175
```

Stage 8: Evaluation and Implementation

Descriptive model: Decision tree model - Model Prediction

- **The decision tree model has many configuration options. One of the most important is the depth of the tree.**
- **However, in the search for the optimal depth level, two phenomena occur over-fitting and under-fitting.**
- **A vital consideration is the accuracy of the data estimated from the validation data. Therefore, the objective is to determine the optimal value between under-fitting and over-fitting.**
- **Different alternatives exist to control the level of tree depth, and many allow some paths through the tree to have a greater degree of depth than others.**

Stage 8: Evaluation and Implementation

Descriptive model: Decision tree model - Model Prediction

However, the max_leaf_nodes argument provides a very sensible way of controlling over-fitting versus under-fitting.

In addition, a utility function can be applied to help compare the MAE scores of different values for max_leaf_nodes.

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode', index_col=0)
pd.options.display.max_rows = None

df2 [ "INGRESOS_LABORALES" ]  =  df2 [ "INGRESOS_LABORALES" ] . astype ( float )
df2 [ "N_PERSONAS_HOGAR" ]  =  df2 [ "N_PERSONAS_HOGAR" ] . astype ( float )
df2 [ "HORAS_LABORALES_SEMANA" ]  =  df2 [ "HORAS_LABORALES_SEMANA" ] . astype ( float )
df2 [ "EDAD_ANOS" ]  =  df2 [ "EDAD_ANOS" ] . astype ( float )
df2 [ "ANOS_ESCOLARIDAD" ]  =  df2 [ "ANOS_ESCOLARIDAD" ] . astype ( float )
df2 [ "N_HIJOS" ]  =  df2 [ "N_HIJOS" ] . astype ( float )
df2 [ "ESTRATO_VIVIENDA" ]  =  df2 [ "ESTRATO_VIVIENDA" ] . astype ( float )

df2["Ingreso_Promedio"] = df2["INGRESOS_LABORALES"] / df2["N_PERSONAS_HOGAR"]
df2.columns = df2.columns.str.strip()
df2["Ingreso_Promedio"] = pd.to_numeric(df2["Ingreso_Promedio"], downcast="float")

df2_Final = df2.loc[ : , ['INGRESOS_LABORALES', 'N_PERSONAS_HOGAR', 'HORAS_LABORALES_SEMANA', 'N_HIJOS',
                          'Ingreso_Promedio','ESTRATO_VIVIENDA'  ]]
print(df2_Final.info())
```

```
from sklearn.metrics import mean_absolute_error
from sklearn.tree import DecisionTreeRegressor
def get_mae(max_leaf_nodes, train_X, val_X, train_y, val_y):
    model = DecisionTreeRegressor(max_leaf_nodes=max_leaf_nodes, random_state=0)
    model.fit(train_X, train_y)
    preds_val = model.predict(val_X)
    mae = mean_absolute_error(val_y, preds_val)
    return(mae)
```

Stage 8: Evaluation and Implementation
Descriptive model: Decision tree model - Important limitations of MAE

The MAE is a measure of the average or the square root of that average of the test error realizations. Error is a numerical random variable and one cannot capture the entire behavior of a random variable with a single aggregation of observations.

Error is just a random variable, often a highly biased random variable. When we predict biased outcomes, such as prices, revenues, item sales, and many more, the error is most likely to be biased as well, which means that in most cases the error is very small, but there are relatively few examples that can have extremely large errors. When the error is highly skewed, the average often says nothing (Gonzalez, 2018).

```
# compare MAE with differing values of max_leaf_nodes
for max_leaf_nodes in [5, 50, 500, 5000]:
    my_mae = get_mae(max_leaf_nodes, train_X, val_X, train_y, val_y)
    print("Max leaf nodes: %d  \t\t Mean Absolute Error:  %d" %(max_leaf_nodes, my_mae))
```

```
Max leaf nodes: 5     Mean Absolute Error:  0
Max leaf nodes: 50    Mean Absolute Error:  0
Max leaf nodes: 500   Mean Absolute Error:  0
Max leaf nodes: 5000  Mean Absolute Error:  0
```

Stage 8: Evaluation and Implementation
Descriptive model: Random Forests

Taking into account the previous results, the random forest model is applied.

The random forest model applies many trees in its internal structure. It makes predictions through the average of the individual predictions of the trees that make it up.

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode', index_col=0)
pd.options.display.max_rows = None

df2 [ "INGRESOS_LABORALES" ]  =  df2 [ "INGRESOS_LABORALES" ] . astype ( float )
df2 [ "N_PERSONAS_HOGAR" ]  =  df2 [ "N_PERSONAS_HOGAR" ] . astype ( float )
df2 [ "HORAS_LABORALES_SEMANA" ]  =  df2 [ "HORAS_LABORALES_SEMANA" ] . astype ( float )
df2 [ "EDAD_ANOS" ]  =  df2 [ "EDAD_ANOS" ] . astype ( float )
df2 [ "ANOS_ESCOLARIDAD" ]  =  df2 [ "ANOS_ESCOLARIDAD" ] . astype ( float )
df2 [ "N_HIJOS" ]  =  df2 [ "N_HIJOS" ] . astype ( float )
df2 [ "ESTRATO_VIVIENDA" ]  =  df2 [ "ESTRATO_VIVIENDA" ] . astype ( float )

df2["Ingreso_Promedio"] = df2["INGRESOS_LABORALES"] / df2["N_PERSONAS_HOGAR"]
df2.columns = df2.columns.str.strip()
df2["Ingreso_Promedio"] = pd.to_numeric(df2["Ingreso_Promedio"], downcast="float")

df2 Final = df2.loc[ : , ['INGRESOS_LABORALES', 'N_PERSONAS_HOGAR', 'HORAS_LABORALES_SEMANA', 'N_HIJOS',
                          'Ingreso_Promedio','ESTRATO_VIVIENDA'  ]]
print(df2_Final.info())
```

```
# Filter rows with missing values
filtered_data = df2_Final.dropna(axis=0)
# Choose target and features
y = df2_Final.ESTRATO_VIVIENDA
df2_features = ['INGRESOS_LABORALES', 'N_PERSONAS_HOGAR', 'HORAS_LABORALES_SEMANA', 'N_HIJOS','Ingreso_Promedio'  ]
X = df2_Final[df2_features]
from sklearn.model_selection import train_test_split
# split data into training and validation data, for both features and target
train_X, val_X, train_y, val_y = train_test_split(X, y,random_state = 0)
```

Stage 8: Evaluation and Implementation
Descriptive model: Random Forests

This model generally has a higher percentage of accuracy in its predictions than the previously implemented model, which consists of a single tree.

The corresponding implementation is presented below.

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error
forest_model = RandomForestRegressor(random_state=1)
forest_model.fit(train_X, train_y)
melb_preds = forest_model.predict(val_X)
print(mean_absolute_error(val_y, melb_preds))
print(forest_model.predict(X))
```

```
0.7851020532577964
[2.19423338 4.49       2.41517847 ... 2.25640396 2.18716162 2.90946216]
```

Stage 8: Evaluation and Implementation
Descriptive model: Random Forests

As can be seen, this value presents a much higher accuracy with respect to the original implementation of the tree model whose initial MAE was 0.4853.

Likewise, it is possible to observe an important variation in the values of the variable ESTRATO_VIVIENDA corresponding to the original data set provided by DANE and the prediction values generated through the two implemented models.

In the three cases analyzed, the predicted values were always different from the original ones.

Stage 9: Feedback

- **The two models developed throughout this project were implemented in a real production environment.**
- **The descriptive model and the predictive model were able to meet all the objectives proposed at the beginning of this work. It was also possible to provide a comprehensive answer to the formulation of the problem: How can data science and ML tools become a fundamental instruments for the generation of socio-economic policies in Colombia?**
- **The impact and scope of this project go beyond the initially proposed frameworks and can become a fundamental consultation tool for the implementation of public policies in different areas as well as in different countries and regions.**

Stage 9: Feedback

- **Likewise, this document becomes a relevant guide both for data science and ML professionals and for citizens with basic programming knowledge, as all the stages involved in the creation of a data science project are addressed and each one of them is explained and analyzed in depth.**
- **Another relevant factor to take into account is that this project was carried out 100% with Open Source tools, which will allow its massive reproduction.**
- **Although the scope of the predictive model was greater than originally established, there are more powerful tools that will increase the efficiency of the models implemented in this work.**

Conclusions

Conclusions

- **The socio-economic stratification instrument does not correspond to the social classification presented by DANE.**
- **The average income per household does not correspond in a high percentage with the range of socio-economic stratification to which they were classified.**
- **The instrument of socio-economic stratification is not a targeting instrument that allows the identification of the poor and vulnerable population of the country, therefore, the subsidies and social programs generated by the government do not impact the neediest citizens, violating the constitutional principle of solidarity with respect to income distribution.**

Conclusions

- **The tools corresponding to emerging technologies such as data science and ML are fundamental instruments for the generation of any type of public policy in any area or region of the world and must become an ally of governments, public entities, and organizations involved.**
- **Data science and ML tools democratize knowledge, empowering citizens, especially the poor and vulnerable populations, by allowing access to and analysis of relevant information for the construction of public policies that have an impact on them. In this way, they are able to contrast and even argue against the figures provided by governments and demand the generation of policies that lead to minimizing the inequality and injustice figures that are present in the country.**

Conclusions

- **Education is the most powerful weapon for the transformation of humanity, and emerging technologies have the potential and the ethical responsibility to become the key instrument for reducing inequality not only in Colombia but also in the Latin American region.**
- **The shortcomings presented in the targeting instrument; socio-economic stratification is a fundamental factor in income inequality in Colombia, which is the highest among all OECD countries and the second highest among 18 LAC countries.**
- **The three predictive models implemented corroborated the findings obtained in the descriptive model; the targeting instrument: socio-economic stratification presents insurmountable shortcomings which had already been analyzed and warned by different groups of experts and international organizations, however, this work allows corroborating and supporting its conclusions through the implementation of data science and ML tools.**

Bibliography

Bibliography

- DANE. "Gran Encuesta Integrada de Hogares - GEIH - 2021". 2022. Directorate of Methodology and Statistical Production - DIMPE. Colombia.
- International Bank for Reconstruction and Development and World Bank. 2021." Towards the construction of an equitable society in Colombia". Washington, DC.
- DIAN and OECD. 2021. "Informe de la comisión de expertos en beneficios tributarios". Colombia.
- Mina, Rosero, Lucia. 2004. "Estratificación socioeconómica como instrumento de focalización". Economía y desarrollo, volume 3 number 1, March 2004. Universidad Autónoma de Colombia. Colombia.
- ECLAC and UNITED NATIONS. 2006. "La estratificación socioeconómica para el cobro de los servicios públicos domiciliarios en Colombia ¿Solidaridad o focalización?". Álzate, Maria, Cristina. Studies and perspectives series. ECLAC Office Bogotá Colombia.
- National Planning Department Republic of Colombia. 2008. "Evaluation of socio-economic stratification as an instrument for classifying users and a tool for allocating subsidies and contributions to household public services. Institutional report and diagnostic report. Bogotá D.C.
- BONILLA, J., LÓPEZ, D., and SEPÚLVEDA, C.E. Socioeconomic stratification and cadastral information. Introduction to the problem and future perspectives. In: Sepúlveda Rico, C.E., López Camacho, D., and Gallego Acevedo, J.M., eds. Los límites de la estratificación: en busca de alternativas [online]. Bogotá: Editorial Universidad del Rosario: Alcaldía Mayor de Bogotá. Bogotá D.C.
- Cerquera-Losada, O., Gómez-Segura, C. and Rojas-Velásquez, L. (2021).” Probability of obtaining an academic degree in Colombia. Educación y Humanismo”23(41),96-118. https://doi.org/10.17081/eduhum.23.41.4105Bogliacino, Francesco, Laura María Jiménez Lozano, and Daniel Reyes Galvis. 2015. "Identifying the Effect of the Socio-Economic Stratification on Urban Segregation in Bogotá." Investigaciones y Productos CID 24, Centro de Investigaciones para el Desarrollo, Universidad Nacional de Colombia, Bogotá.
- DANE. 2021. "Manual de recolección y conceptos básicos gran encuesta integrada de hogares." Colombia.
- DANE. 2022. "Analysis of social classes in the 23 cities and metropolitan areas of Colombia 2019 - 2021". Colombia.
- OECD and EC (European Commission). 2020. "Cities in the World: A New Perspective on Urbanization." Highlights, OECD Urban Studies, OECD and EC, Paris and Brussels.
- Banco de la República (Bogotá). 2014. Economics of large cities in Colombia: six case studies / Banco de la República, Gerson Javier Pérez et al. -- Editor Luis Armando Galvis. -- Bogota: Banco de la República.

Bibliography

- OECD (Organization for Economic Co-operation and Development). 2014. "OECD Territorial Reviews: Colombia 2014." Paris, OECD.
- OECD and World Bank. 2012. Tertiary Education in Colombia. Reviews for National Policies for Education. Paris and Washington, DC: OECD and World Bank.
- Bernal, Raquel, Marcela Eslava, and Marcela Meléndez. 2015. "Taxing Where You Should: Formal Employment and Corporate Income vs. Payroll Taxes in the Colombian 2012 Tax Reform." Universidad de Los Andes, Bogotá.
- Campos, Ana, Niels Holm-Nielsen, Carolina Díaz, Diana M. Rubiano, Car - los Costa, Fernando Ramírez, and Eric Dickson. 2012. "Analysis of Disaster Risk Management in Colombia: A Contribution to the Creation of Public Policies." World Bank, Washington, DC.
- CONPES (Consejo Nacional de Política Económica y Social República de Colombia/Nationall Council for Economic and Social Policy of Colombia). 2009. "Lineamientos para la consolidación de la Política de Mejoramiento Integral de Barrios (MIB)." Documento CONPES 3604, Departamento Nacional De Planeación, Bogotá.
- Del Carpio, Ximena V., José A. Cuesta, Maurice Kugler, Gustav Hernández, and Gabriel Piraquive. 2020. "Equity Aspects of Jobs and Economic Transformation (JET) in Colombia: What Effects Could Global Value Chain and Digital Infrastructure Development Policies Have on Poverty and Inequality after COVID-19?" Background paper for this report. Unpublished.
- EC (European Commission). 2019. Key Competences for Lifelong Learn - ing. Directorate-General for Education, Youth, Sport, and Culture. Brussels: European Commission. Echavarría, Juan José, Iader Giraldo, and Fernando Jaramillo. 2019. "Global Value Chains, Growth and Tariff Protection in Colombia." Borradores de Economía 1080, Banco de la Republica Colombia, Bogota.
- Cifuentes, Valerie. 2021. "This is what the richest 1% in Colombia contributes in taxes." Revista Forbes Colombia.
- Esguerra, Maria del Pilar, and Sergio Parra Ulloa. 2016. "Colombia, Outside Global Value Chains: Cause or Symptom of Low Export Under-performance?" Borradores de Economía 966, Banco de la Republica Colombia, Bogotá.
- Meltzer, Joshua Paul, and Camila Pérez Marulanda. 2016. "Digital Colombia: Maximizing the Global Internet and Data for Sustainable and Inclusive. Growth". Documento de trabajo 96, Global Economy and Development, Brookings Institution, Washington, DC.
- Eslava, Marcela, John Haltiwanger, Adriana Kugler y Maurice Kugler. 2004. "The Effects of Structural Reforms on Productivity and Profitability Enhancing Reallocation: Evidence from Colombia". Journal of Development Economics 75 (2): 333-71.

Special Thanks!!!

Special Thanks

- **Firstly, I would like to express my gratitude to the entire KaggleX BIPOC project team for this life-transforming and community-transforming initiative. All their dedication, effort, and support in providing us with incredible tools for training and growth have borne fruit through the creation of a global community united by the love of data science and ML, constantly supporting each other and focused on generating high-impact projects and developments.**
- **Secondly, I would like to thank my mentor Sagar Ganapaneni, a source of inspiration in my professional development, whose guidance and ongoing support enabled me to exceed the goals set at the beginning of this program. Special thanks also to my mentor Mani Sarkar who was always there to support us all, for taking the time and effort to take us as his own mentees. Thanks to the team of mentors who were always willing to guide us and share their knowledge regardless of schedules, their commitment to our development was fundamental in the process.**
- **I would like to extend my thanks and appreciation to all my colleagues, for their constant support, their continuous willingness to create community, and of course for all their efforts to successfully complete this program.**

Let’s keep in touch

- www.linkedin.com/in/claudia-isabel-reyes-moreno-30a244106
- https://gdg.community.dev/gdg-cloud-sabana/
- https://github.com/CLAREISMO
- https://www.kaggle.com/clareismo

Presen

Presenter Name

Presen

Presenter Name

Presen

Presenter Name

Presen

Presenter Name


kaggle