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

Claudia Isabel Reyes Moreno

***Abstract*—This paper develops the conceptual framework under which the project "Approach to the application of Data Science and ML as key tools for the construction of public policies in Colombia" is based, whose main objective establishes the study and analysis of various socio-economic variables that influence and determine the generation of state policies. Throughout this article we will develop the state of the art, the problem statement, the background, the justification of the project, as well as the theoretical and methodological framework of the project, giving way to the implementation of the analysis corresponding to the selected data set; which is constituted by data obtained from the following sources: socio-economic information on Colombia updated to 2021 by the World Bank, DANE (Colombian National Department of Statistics), Economic Commission for Latin America (CEPAL), Kaggle Portal as well as by the Colombian government's open data portal. This analysis will be implemented through data science and Machine learning tools to generate conclusions and proposals that can become a base instrument for the generation of policies that have a real and deep impact on vulnerable populations in Colombia.**

1. INTRODUCTION

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. Adult income disparities arise from gaps that open up early in life for high-quality opportunities in child development, education, and health care services. Inequality in access to quality jobs further amplifies these gaps, making Colombia one of the countries where inequalities are most persistent between generations. Long-standing inequality between regions overlaps with significant gaps in well-being between Afro-descendants and indigenous Colombians and the rest of the population. The COVID-19 pandemic has further amplified existing disparities and threatens to have prolonged adverse effects; but this is only one of many potential extreme shocks, including climate change-related shocks, that could substantially widen inequality gaps. At best current tax and transfer policies, at best, have only a modest positive impact on these imbalances, so there is ample potential to improve the redistributive role of fiscal policy in Colombia. Policy reforms in many areas could help chart a more equitable future for the country (World Bank Group, 2021).

2. PROBLEM STATMENT

How can data science and ML tools become a fundamental instruments for the generation of socio-economic policies in Colombia?

3. BACKGROUND

The incorporation of new technologies for the management and analysis of large volumes of data from the IoT, drones, and satellite data applied to agriculture, gave rise to Smart Farming (Smart or precision agriculture), which allows, for example, water savings by establishing irrigation times more in line with the progress of planting, which translates into higher profits and lower production costs (Deutsch, 2018). In Latin America and the Caribbean, a 2017 study identified 130 innovation ventures using technologies such as Big Data, IoT, ML, and biotechnology, focused on the agriculture sector. Among them, more than 60% were created in the last five years, mostly in Argentina, Brazil, Chile, and Colombia (Fontagro, 2018).

On the other hand, it is expected that this proliferation of massive data will serve as food for new scientific and innovation developments that, supported by technologies such as Big Data and Blockchain, will contribute to a more efficient and sustainable development (CERAI - Center for Rural and International Agriculture Studies, 2019), and will influence the entire supply chain by driving predictive processes and the redesign of business activities among the different actors involved (Wolfert et al., 2017).

The following is a compilation of the most relevant implementations in Colombia.

A. Agricultural sector

About this aspect, in Colombia, Hernández (2016) developed a model for the management and analysis of environmental data with the use of Big Data and AI that will support decision-making by enabling the processing and storage of meteorological and hydro-meteorological data generated by environmental monitoring stations, to then apply Deep Learning in predicting the behavior of variables such as precipitation, temperature, humidity, and pressure. Examples of Smart Farming and data analytics include the CGIAR (Consultative Group on International Agricultural Research) Platform for Big Data in Agriculture, managed by CIAT (International Center for Tropical Agriculture) in Colombia (Lazo Cardona,2019).

In Colombia, the Farm app technology platform is helping the agricultural sector. Farm app implements Big Data, IoT and geolocation tools, and satellite technology, to monitor crops for pests, make forecasts because of the impact of climate change, and analyze the efficiency of pesticides and soil conditions to make a more appropriate planting (Asociación Nacional de Industriales, 2017).

B. Education sector

In the education sector, some informatics platforms are collecting large-scale data on teaching-learning activities, of which there were no precedents, applying Data Mining and AI techniques to improve educational quality through data analysis (Raffaghelli, 2020). This has given rise to a new discipline called Learning Analytics (LA), whose fundamental objectives are the reflection and prediction of massive data (structured and unstructured), through social processing with analytical tools such as ML or classical statistical analysis (Rojas-Castro, 2017).

Several are the researches that apply LA and Big Data as tools to strengthen teaching-learning processes. For example, at the National University of Colombia, Manizales branch, a master's degree work proposes the development of a pattern discovery model through data mining and LA, which was fed with educational data and student interactions with virtual learning platforms, to improve the teaching-learning process making it adaptive to the particularities of each student (Giraldo-Ocampo, 2017).

On the other hand, the analysis of student data could also serve to combat academic dropout and increase quality standards in teaching. This is the case in Colombia, where a research project at the Universidad Distrital Francisco José de Caldas proposes a model for analyzing massive data resulting from academic processes and relates them to historical and personal data to combat academic dropout (Rodríguez et al., 2019).

C. Tourism sector

In the tourism sector, data analysis tools such as Big Data, together with the implementation of data analytics algorithms and AI, are marking a new path in the promotion and dissemination of tourism centers in Latin America. In the Colombian context, a study used data science applied to geographic

information systems to analyze and visualize different variables relevant to ecotourism, to identify deficiencies and opportunities and improve decision-making in the tourism sector (Barrera et al., 2020). Likewise, the Tourism Sector Plan 2018-2022 of the Ministry of Commerce, Industry and Tourism established that for the tourism sector, it is a priority to be at the forefront of new technologies that identify preferences and consumer behavior patterns, to improve marketing strategies and decision making through data analysis, Big Data, AI and ML (Ministry of Commerce, Industry and Tourism, 2018).

D. Public management

Concerning public management New data storage and analysis technologies are changing the governance of countries; despite the great benefits they offer by providing more efficient mechanisms to provide services to their citizens, they are still not government plan strategies for many developing countries. Regarding this area in Colombia, some efforts are presented as is the case of the Comptroller's Office of Bogota, which used Big Data to conduct an audit process in which it found fiscal findings for approximately US$72,000 (OLACEFS, 2018). On the other hand, the MinTIC has promoted the use of Big Data with the National Planning Department to execute an update of national GDP variations; it has also detected fraud in the policies of the System of Identification and Classification of Potential Beneficiaries for social programs (SISBEN) by tracking 653,000 cases of inconsistencies in the system in 2015 (Gomis-Balestreri, 2017).

The adoption of open data strategies has been being promoted in the region thanks to the creation of the Pacific Alliance (PA), made up of Chile, Colombia, Mexico, and Peru, where one of the principles of the fourth pillar "Digital Government" is the adoption of Open Data as a tool to promote evidence-based public policies, encourage citizen participation, bring state institutions closer together and stimulate economic growth (Coordination of National Digital Strategy, Government of Mexico, 2017).

E. Industry and economy sector

In the area of Industry and Economy, Big Data and data analytics are tools that help organizations in the decision-making process by improving segmentation, communication, and customer loyalty, which is vital as a competitive advantage in an increasingly globalized market. The digitization of the industrial sector is stimulating the data center market in Latin America. This is expected to grow by 6% for the period 2019-2025, for factors such as the development of green data centers, smart agriculture, and e-Commerce to be the highest usage of Big Data analytics and IoT applications (Lazo Cardona,2019).

According to a recent study, the Latin American Big Data and Analytics (BDA) market reached revenues of US$2,992.5 million in 2017. It is expected to generate US$8,593.5 million by 2023. Currently, Brazil leads the ranking with 46.7% of total sales, followed by Mexico (26.7%), Colombia (7.9%), Chile (6.9%), Argentina (5.6%), and Peru (2.4%) (Frost and Sullivan, 2018).

In this context, the Economic Commission for Latin America and the Caribbean (ECLAC) is leading the project "Big Data: Big Data for the Digital Economy in Latin America and the Caribbean", which aims to improve capacities for measuring the digital economy and designing evidence-based policies through big data analytics. In addition, it seeks to analyze the Internet digital economy in Brazil, Chile, Colombia, and Mexico, based on the processing of web and official data with Big Data (ECLAC, 2017).

On the other hand, with the help of Big Data, in Colombia, it is possible to perform consumption analysis (Voice-Data) generated by calling on your cell phone or mobile internet consumption to generate marketing strategies in real-time, according to customer preferences (Soche, 2016).

Another national example shows how the Colombian Ministry of Finance and the DNP developed a methodology, based on Google Trends data, to statistically analyze search terms to predict and obtain economic trends in certain sectors faster than through traditional statistics (Karisma, 2016).

F. Health sector

One of the sectors that most benefits from implementing data collection and analysis techniques are health, because those tools contribute to the reduction of medical research costs, and serve to find predispositions and

symptomatic patterns in various diseases. This results in better diagnosis and treatment and boosts research into new drugs. This is what happened with the company Berg, which used AI to discover the drug BPM 31510 to fight cancer (Guillén, 2017).

Thus, Big Data can help to discover the danger of a pandemic by identifying real time trends in Internet search engines such as Google Trends or data systems (Monleón-Getino, 2015). A clear example of this methodology occurred in Colombia, where research developed between the ISI Foundation and the United Nations describes that it can associate the contagion of Zika between 2014 and 2016 through the tracking of the geographical position of people when using their cell phone (Perrotta, 2018).

In Colombia, the department of Cundinamarca implemented the Unified Electronic Health Record (HCEU), a technological transformation project, which seeks to optimize and integrate the different information resources of the health services in the department by unifying the clinical history in the 35 Hospitals of the Departmental Public Network, facilitating evidence-based medical decision making (Gobernación de Cundinamarca, 2018).

G. Challenges in the implementation of technological tools

According to the above, it is possible to infer that Colombia presents an awakening toward the application of technological tools such as data science, ML, and AI as fundamental instruments in the exploration, treatment, and analysis of data. However, it can be seen that the greatest contributions come from projects generated by the educational sector, specifically by universities at the graduate level and by the impulse generated by international organizations such as the Economic Commission for Latin America and the Caribbean (ECLAC), the Pacific Alliance (PA), the Organization for Economic Cooperation and Development (OECD), of which Colombia is an active member through the application of the open data strategy.

There are also some projects coming from the public sector, without establishing general guidelines for the adoption of these tools, added to the delay that the country presents concerning the digitization of documents which is at a very low level n "Artificial Intelligence will make people more efficient. It is necessary to use structured data, but only 10% of this data has this condition. Companies need to be transformed, they need to be digitized. This starts by facilitating infrastructure and seeking more productive models through technology," said Felipe Villa, corporate director of Digital Transformation at Claro Colombia.

A clear example of this problem was evidenced by the paralyzation of judicial services in Colombia during the health situation caused by the pandemic. This was demonstrated in the tutela ruling with which a court in Pasto suspended a virtual public hearing on the return of glyphosate, scheduled for May 27, because it was for communities that do not have internet access and, therefore, could not participate in it.

As Hernando Herrera, director of the Corporación Excelencia en la Justicia (Excellence in Justice Corporation), says, the country is still in the prehistory of digital justice, which prevents it from having more agility, transparency, and interoperability.

Colombia also has a large backlog of infrastructure and internet connectivity. Only 56.5% of households in the country have internet access, although the figure is low for Colombia as whole, rural areas are the worst off as only 23.8% of households in these territories have internet access. However, when reviewing the figures by strata, the gap is evident. While 21% of households in stratum one have internet access, 99.8% of households in stratum six are connected (DANE, 2021). Even at least 2.5 million Colombians do not have electricity in their homes. The Colombian Association of Electric Power Generators (Acolgen) has indicated that some 470,000 homes do not have access to electricity services in Colombia. With an average of four people per household, it is estimated that at least 2.5 million Colombians live without electricity.

In addition, the Survey of Information and Communication Technologies in Homes (ENTIC Hogares, 2021) of the Dane revealed that only 39.3% of households in Colombia have a computer, the analysis found that the main reason why households did not have a computer in 2020 is that it is too expensive, both nationally and in dispersed rural and populated centers with proportions above 58.0%. Thus, in the year in which the Coronavirus pandemic demanded digitalization, 60.7% of Colombian households did not have a computer and 24.6% did not have a cell phone (Portafolio, 2021).

Currently, in the market, there is a low number of personnel trained in digital skills, which is an unprecedented problem. A study by EAFIT-Infosys shows that in 2018 the country had a deficit of 35,504 Information Technology (IT) professionals in a scenario of low growth in demand, but this deficit could reach 94,000 professionals in a scenario of high growth, in the figure some branches such as artificial intelligence, cybersecurity, and data science stand out. According to the high demand for professionals specialized in digital skills that the country has and the deficit of human capital in these areas, especially in data analysis, it is necessary to train more people in data science and artificial intelligence, to provide trained personnel to companies (Ministry of Telecommunications, 2020).

With the above mentioned, Colombia faces great challenges in the adoption of the technological tools mentioned above, especially in the areas of public management. No evidence of previous works with the general objective covered by this project was found, therefore this work is the first approach to the analysis of these datasets through concepts related to data science and ML.

4. JUSTIFICATION

The presence of the State and its administrative control of the national territory has been conditioned during the last fifty years in Colombia by the control that rebel and paramilitary groups have exercised in various parts of the country, especially in the most remote regions, with the consequent weakening of local administrative capacity in those areas. Hence, one of the main pending tasks of governance aimed at implementing the national development strategy is strengthening the administrative capacity of subnational governments to formulate and implement public policies and services that reflect national standards and quality, especially in rural and remote areas, in full coordination with the national government (OECD, 2022)

On May 30, 2018, the Republic of Colombia signed an agreement to join the OECD. It will join the Organization as a member country at the time it deposits its instrument of accession to the OECD Convention. As part of its accession process, the country has taken important steps to reduce poverty and improve the delivery of public services to its citizens. The completion of this work has led Colombia to commit to continued post-accession reforms in the areas of public governance identified as priorities by the delegates of the OECD Public Governance Committee. Among its post-accession commitments, Colombia has agreed to pursue reforms in four key areas of public governance: 1. Effectiveness and efficiency of justice institutions. 2. Transparency and accountability. 3. Anti-corruption and integrity frameworks and institutions. Subnational administrative capacity. While the fourth area focuses on improving subnational administrative capacity, the other three relate to both the national and subnational levels (OECD, 2022).

The project funded by the Swedish International Development Cooperation Agency (SIDA) entitled: "Fostering institutional efficiency and public governance effectiveness in Colombia as strategic factors to support inclusive growth and bring Colombia closer to the OECD" in its final report identified the following facts as the main difficulties faced by the Colombian state to generate efficient public policies that positively impact the vulnerable population:

- The departments have several planning instruments that are rarely interrelated and with different execution deadlines, which limits their capacity to coordinate them and to monitor and evaluate the results and consequences on public policy.
- They lack sufficient financial and human resources to lead horizontal coordination between different secretariats or units. This fact constitutes a barrier to coordinating and implementing public policies and evaluating their results in an effective manner.
- There is still limited stakeholder participation throughout the policy cycle, especially in the monitoring and evaluation of the Departmental Development Plans. While interviews conducted have shown that consultation is a common practice in the early stages of plans, active involvement of citizens, NGOs, and representatives of the private sector and the media remains rare in the evaluation phase.
- Few departments include open government principles and initiatives related to transparency, accountability, integrity, and participation in Departmental Development Plans.

Good public governance and sound administrative capacity are essential for the successful implementation of public policies in any area. Good public governance is a sine qua non of pluralistic democracies for the rule of law and respect for human rights. Effective democratic institutions are at their core and an indispensable means for open, equitable, and inclusive decision-making in the public interest and in concert with citizens to enhance the well-being and prosperity of all. In its Policy Framework for Good Public Governance (OECD, forthcoming), the OECD defines this concept as "the formulation, implementation, and evaluation of rules, processes and formal and informal interactions between the institutions and actors that make up the State, and between the State and citizens, either individually, in the form of civil society organizations, businesses, and other non-state actors, that frame the exercise of public authority in the public interest and decision-making that allows for adequate anticipation and detection of problems and, in response, underpins increases in overall prosperity and well-being" (OECD, 2022).

For the project (SIDA) Good public governance is therefore the combination of three interconnected elements:

- Values: key behavioral traits that guide public governance in all its dimensions in a way that promotes and protects the public interest.
- Enabling factors: an integrated web of practices to correctly identify problems and challenges, and to formulate, implement and evaluate reforms to improve performance.
- Instruments and tools: a set of policy instruments and management tools for effective public policy formulation.

According to the Policy Framework, the coordination of all administrative units is one of the main factors contributing to sound public governance. The open government principles of transparency, accountability, integrity, and participation are part of the values of such governance (OECD, 2022).

It is precisely within the framework described above, specifically in the third item "Instruments and tools" where the developments of new technologies such as data science and ML become fundamental instruments within the management tools that allow reorienting the course of the enabling factors by effectively and truthfully detecting the problems, difficulties, populations, and solutions conducive to the equitable development of the Colombian state.

Planning in the public sector is the mechanism through which public policies are developed and implemented in a way that takes into account the needs of civil society and the effective and efficient use of available resources (Friedman, 1987). For the OECD, strategic planning refers to the process of developing, implementing, monitoring, and evaluating a strategic vision for a given department in Colombia. The planning process requires a high degree of coordination and leadership among all administrative nuclei, with other levels of government, and with a broad set of external actors to give a specific form to the vision, ensure its coherence, make it operational, and evaluate its performance against the strategic results it is supposed to achieve.

In 2019 Colombia developed its Digital Transformation and Artificial Intelligence Policy, through CONPES 3975 of 2019. In this document, it was recognized that one of the main phenomena that the country had to face, in terms of artificial intelligence (hereinafter AI), was the information asymmetries between different actors both within the country and internationally. These asymmetries were experienced at different levels. There were marked differences between the private sector's knowledge of this technology and the public sector's knowledge of it. And, beyond the asymmetries, the lack of knowledge that exists about AI in the public sector can lead to its underutilization and the generation of expectations that are not realized or do not correspond to reality. Likewise, it became evident that the academic sector of the country faced important challenges to access the tools and information on the subject, which are available in the great centers of world knowledge (Ministry of Information Technologies and Communications of Colombia, 2022).

In 2021, the Colombian government proposed the realization of the first Expert Mission focused on Artificial Intelligence. Thus, on October 21, 2021, the launching of the Mission of Experts in Artificial Intelligence took place, which became necessary as a mechanism to establish a prospective roadmap to achieve the implementation of this set of technologies from the technical and comprehensive

vision of experts as a complement and guide in the path that Colombia has been tracing. This mission was completed in July 2022. Since October 21, 2021, the mission worked on the development of 3 projects which are implemented in the first phase: i. AI empowerment platform, ii. Public policy lab focused on the future of work and gender and iii. Artificial Intelligence for Sustainable Ecological Development - A Roadmap for Colombia (Presidential Advisory Office for Digital Transformation and Management and Compliance, Presidency of the Republic of Colombia, 2022).

In the document "Recommendations for the development and strengthening of AI in Colombia" generated by the Mission of Wise Men, they refer to the innovation processes in the public sector: "they have not been able to test different models and experiment to scale many of these initiatives. Sometimes they wait to see what works in other countries and then implement these initiatives in the country. This type of knowledge transfer tends to be costly and sometimes does not bear the desired fruits, as it does not obey the specific context of each country. This leads to long delays and countries being left behind in phenomena as transformative as AI. In addition, there is a need to improve access to data, evidence, and information. The information classification systems in Colombia around technology and existing measurements are few. This is something that still needs to be further explored if the country is to take full advantage of these experiences. It is expected that the country will deepen its measures of evaluation and monitoring of technology policies in the coming years, which is only possible if it begins to make significant efforts in the collection and access to information on the progress of emerging technologies in all sectors".

Likewise, the Ministry of Information Technology and Communications of Colombia in its report "Analytics tools for data exploitation 2018" determined that: "The data stored by state entities are generated quickly and are complex, however, within them is contained a large amount of extremely useful information for decision making; that is why it highlights the importance of analyzing and interpreting these data and convert them into information that state entities can use in their favor to improve their performance and be more efficient. Data analysis refers to the review of data to draw conclusions from the information. This activity allows for improving the decision-making of companies, entities, and institutions in general. Data analysis is essential for the fulfillment of the objectives of all government agencies since it allows for obtaining more effective results more efficiently. The growing volume of data and its increasingly diverse nature allow having more enriching information for decision-making, however, they can also paralyze the organization, since a large amount of information has caused the traditional means of storage and processing to be insufficient, and this is where Big Data processing comes in, as it allows the storage and examination of large volumes of data at high speed, to extract patterns and unknown and useful relationships for understanding the operation and improvement of the institution; These improvements translate into the streamlining of decision making for the generation of social and economic value, whether products, services or processes".

For all the above mentioned, the implementation of this work is widely justified and constitutes a basic tool for the analysis of variables conducive to the generation of public policies of the Colombian state.

5.THEORETICAL FRAMEWORK

Colombia is one of the most unequal countries in the world.

Income inequality in Colombia is the highest among all OECD countries and the second highest among 18 LAC countries.1 The Gini coefficient of household income (a standard measure of inequality) reached 0.53 in 2019, after paying taxes and receiving transfers. For comparison, the Gini coefficient of the most equal OECD country, the Slovak Republic, was 0.24. The income of the richest 10% of the Colombian population is eleven times higher than that of the poorest 10%. Again, by way of comparison, in the Slovak Republic, the richest 10% of the population earns three times more than the poorest 10%. The economic impact of COVID-19 has further increased inequality, pushing the Gini coefficient up to 0.54 in 2020 and dragging about 3.6 million more people into poverty. There are also significant inequalities between different population groups. A woman in Colombia is 1.7 times more likely to be unemployed than a man. An indigenous Colombian receives on average two fewer years of schooling than other Colombians, and an Afro-Colombian is twice as likely to live in a poor neighborhood. Two-thirds of the children of migrants

from Venezuela are not enrolled in school, compared to less than one-tenth of non-migrants. Surprisingly, inequality in Colombia extends beyond material aspects of livelihoods (World Bank Group, 2021).

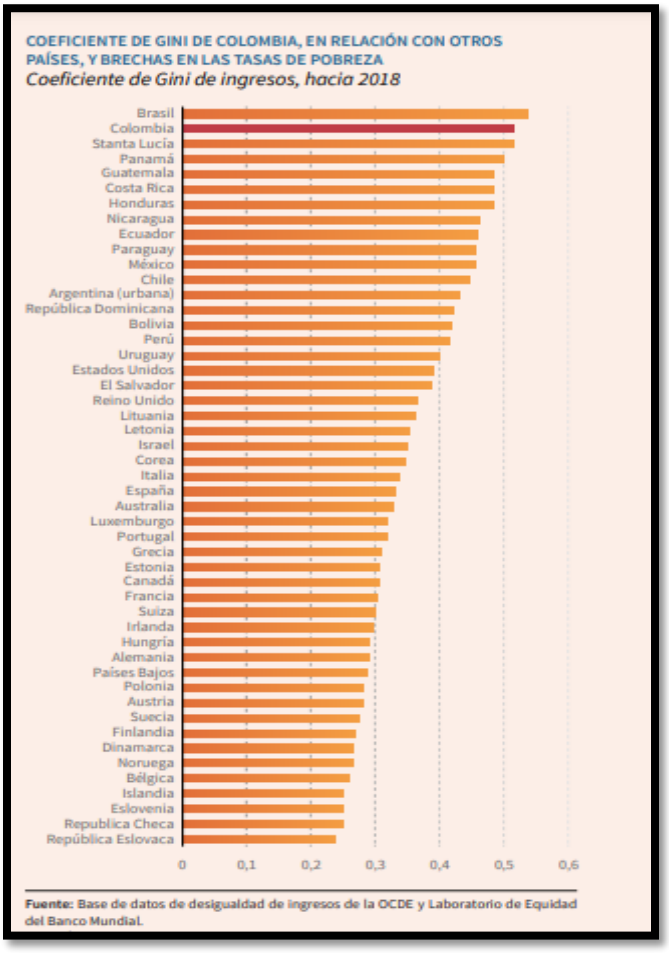


Fig.1. Gini coefficient in Colombia in relation to other countries, and gaps in poverty rates**.**

Colombians with less education, the rural population, and the unemployed or poor are much less likely to consider themselves happy. Inequalities also persist between generations. Children in Colombia face very different life prospects because of the circumstances into which they are born: a child of a low-income parent is likely to earn less than a child of a high-income parent. Among a group of 75 countries, the transfer of the income gap from one generation to the next in Colombia is the most entrenched (Narayan et al. 2018).

Children from poor families could take an average 330 years to move out of poverty, or eleven generations to reach the median income of their country (OECD, 2018). The study "A broken down social elevator? How to promote social mobility", which spanned four years and included more than 20 countries in different parts of the world, points out that social mobility stagnated and inequality increased in the last decade.

Based on this study, Colombia is the most unequal country in Latin America due to its high concentration of income. And this is not the usual measurement using the Gini Coefficient. In the OECD study, the researchers analyzed the household income of the 40% of the population with the least economic resources and the richest 10%. It was there that they observed the gaps in income distribution, but they were particularly struck by the regional differences in the country and the low quality of access to good quality jobs (Gabriela Ramos, Director General of the OECD,2019).

A large part of the population moves into the world of informal employment, with no real possibilities for progression. "They have no medical coverage, no pensions, no basic services. So the difficulties are reproduced because the redistributive impact of the tax and social security system does not reach the poorest", Likewise in several countries of the region there are families that manage to get out of poverty, but quickly fall back into it because of any fortuitous event, something that some analysts often call "a vulnerable middle class" that returns to its original situation at any unforeseen event. For example, it is enough for a member of the family to fall ill for the circle to repeat itself. Just as wealth is inherited, poverty is also inherited (Gabriela Ramos, Director General of the OECD, 2019).

Several governments in Colombia have implemented cash transfers to reduce poverty, but such policies do not change the underlying problem: inequality. And in the case of Colombia, the situation is more acute. In addition, the country has poor results in international tests that measure the quality of education, an essential factor for social mobility. "In Colombia, there are barely 11% of resilient students, which are those who obtain higher grades than their socioeconomic status would predict" (Gabriela Ramos, Director General of the OECD,2019).

This high inequality is rooted in large regional disparities, as there is a large gap between urban and rural areas. Colombia has one of the highest levels of regional inequality in GDP per capita among OECD countries (OECD, 2014). Inequalities particularly affect ethnic minorities and people displaced by conflict, who are disproportionately concentrated in rural areas. Inequality is also a

gender issue, as female employment is low and wage differentials have been increasing.

The increasing flow of immigration from Venezuela mainly affects the northwestern regions of the country, adding to regional disparities. Access to high-quality education and health is also unequal across regions and socioeconomic groups The pension system exacerbates inequalities, leaving many older people in poverty, given the low coverage among the most vulnerable population (OECD, 2019).

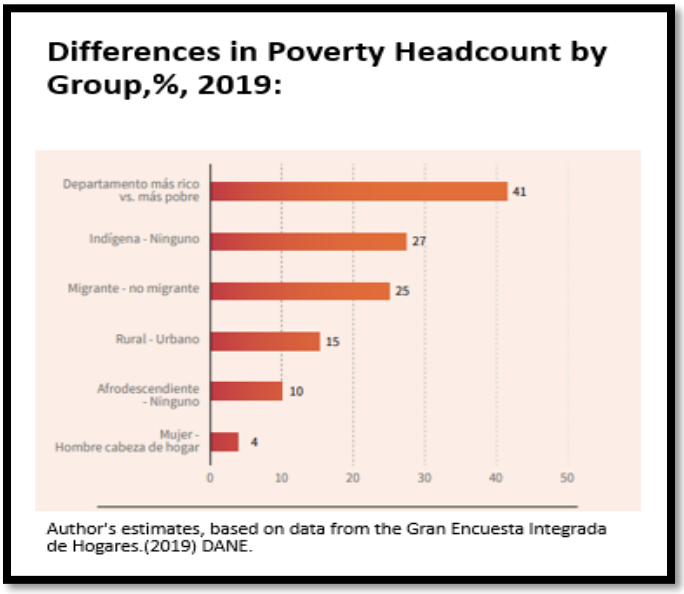


Fig2.Differences in Poverty Headcount by Group,%, 2019:

Inequalities begin in early life, with gaps in education and health care.

The report "Towards building an equitable society in Colombia" implemented by the World Bank in 2022 determined that in Colombia, inequalities affect people from the beginning of their lives in a way that has consequences on the accumulation of human capital and, therefore, on the opportunities available when entering the labor market or earning an income. First, learning opportunities are not the same for all children in Colombia.

Afro-Colombian and indigenous populations lose the equivalent of 4.7 and 4.5 years of education, respectively, when adjusting years of schooling with actual learning outcomes, figures that exceed what other groups lose by a full year or more.

The gaps between schooling and learning are largely related to differences in teacher quality and the way teachers are assigned to schools.

The study also found that despite a substantial expansion in health insurance in recent years, there are large differences in access to high-quality medical care. This contributes to disparities in outcomes: poorer children have stunting rates that are three times higher than those of wealthier children. Two factors affect the quality of health care delivery:

- Current formulas for financing health care providers do not take into account the risk profile of the typical patient, providing little incentive to extend differentiated care to patients with different risk factors.
- Information on the quality of service delivery is limited and insufficiently used to better plan and deliver services at the local level.

Promoting human capital accumulation from early childhood requires simplifying administrative procedures for citizens to access early childhood development (ECD) services, introducing a core curriculum for basic skills throughout the education system, and providing pedagogical support to teachers on the core curriculum guidelines.

At the same time, it also requires strengthening the linkages between basic and tertiary education and ensuring the quality and relevance of the curriculum. In health, the service delivery model must be transformed into a primary healthcare system that is adapted to local needs, and accreditation and financial incentives must be provided to health insurers. In gender, for example, there is a pending agenda to address the barriers in labor regulations that affect women as well as the few policies generated to increase access to and quality of childcare, so that more women can participate in the labor market (World Bank Group, 2021).

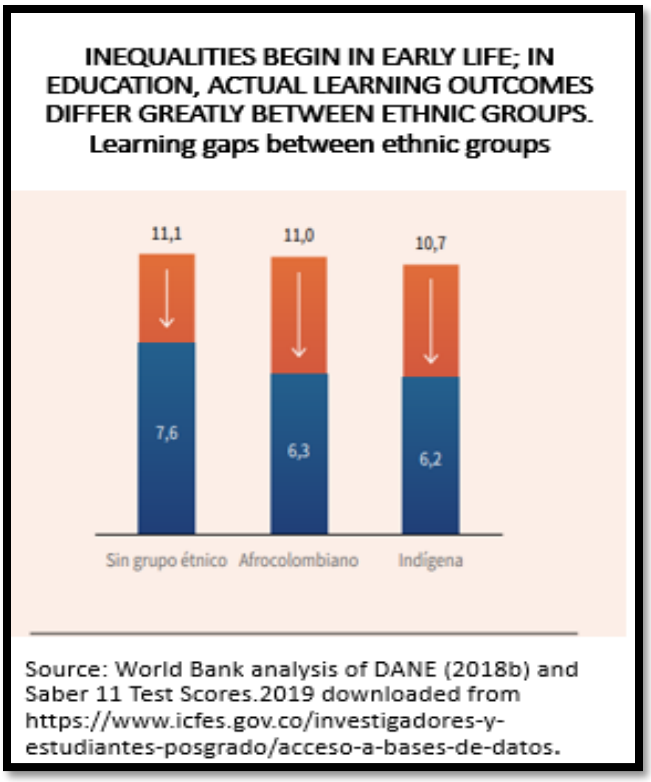


Fig3 Inequalities begin in early life; in education, actual learning outcomes differ greatly between ethnic groups. learning gaps between ethnic groups.

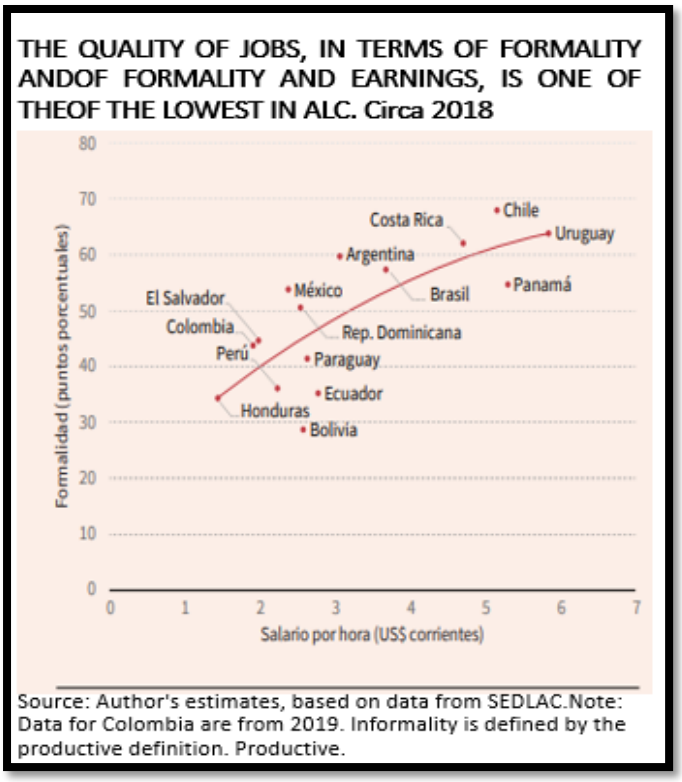


Fig4. The Quality of Jobs in Terms of Formality and Earnings in LAC.

Disparities in access to good jobs widen inequalities in human capital.

Only 40% of working Colombians are employed in the formal sector, one of the lowest rates in LAC. There are no policies aimed at creating jobs in the formal sector, leaving the majority of Colombians working in the informal sector.

The jobs of the future may also be out of reach for many, due to the slow adoption of new technologies among disadvantaged groups, a barrier that the COVID-19 crisis has exacerbated.

Indeed, with a ranking of 109 out of 141 countries, Colombia has one of the world's largest disparities in technology use among socioeconomic groups: while 73% of people in the top 60% use the Internet, that figure is only 53% among those in the bottom 40% (World Bank, 2021).

Territorial inequalities are also high, leaving many people disconnected from key services and opportunities.

The gap between the richest and poorest regions in Colombia is more than double that of other OECD countries5. Spatial disparities overlap with population groups defined by ethnicity: municipalities with high concentrations of indigenous Colombians have persistently high levels of unmet basic needs, and Afro-Colombians live predominantly in urban areas where, between 2005 and 2018, unmet basic needs remained above other cities. Inadequate housing and infrastructure are the main sources of inequality in cities.

Of the 5.1 million households with housing vulnerability in Colombia6, almost 4.4 million are in urban areas. Inequality has worsened in the territories most affected by the armed conflict, which has intensified disparities in access to productive factors, particularly rural land.

Colombia is among the top five most unequal countries in the world, in terms of land concentration (Cuesta and Pico, 2020b): 81% of private land is concentrated in the top 1% of farms, the highest

among the 15 countries in the region, and significantly higher than the regional average of 52% (Guereña, 2017).

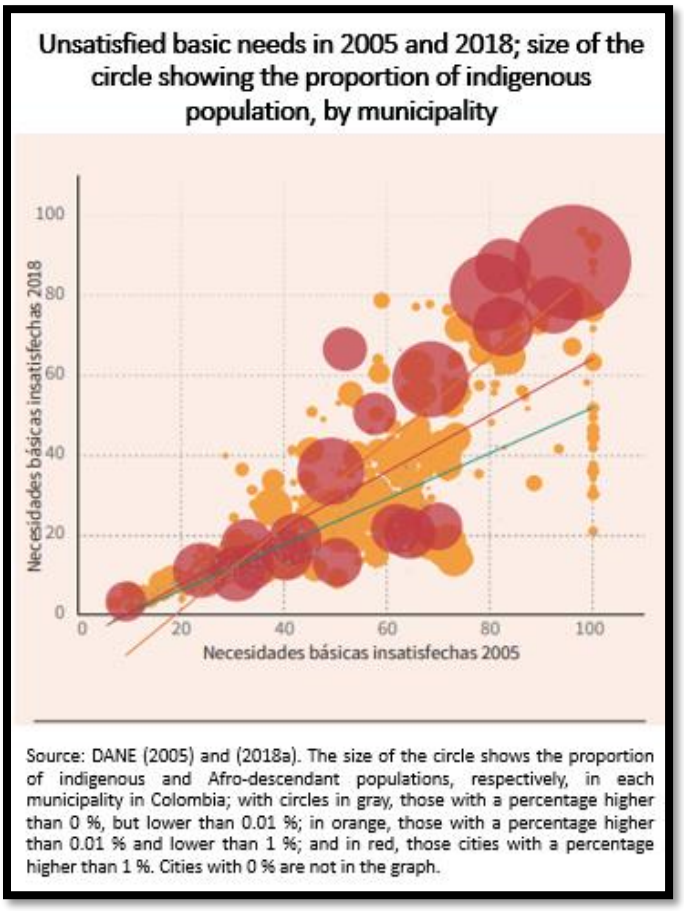


Fig5. Unsatisfied basic needs in 2005 and 2018.

Reducing territorial inequality requires policies that strengthen the technical capacity and fiscal performance of subnational governments, particularly among those that lag and need more support. Expanding connectivity, from residential sections of peri-urban areas and smaller municipalities to the tertiary and secondary road network, and strengthening housing programs can also increase access to opportunities and reduce inequalities.

Policies should better target those population groups that have historically been segregated (e.g., Afro-descendants, indigenous communities, and, recently, migrants) (World Bank, 2021).

Taxes and transfers do little to address glaring inequalities.

Compared to other OECD and LAC countries, taxes and transfers in Colombia do little to reduce income inequality. Because deductions and tax thresholds in the personal income tax (IRP) are very high, people start paying it only if their income is very high, about four times the median income. This deprives the state of resources that could be redistributed to the poorest (World Bank, 2021).

.

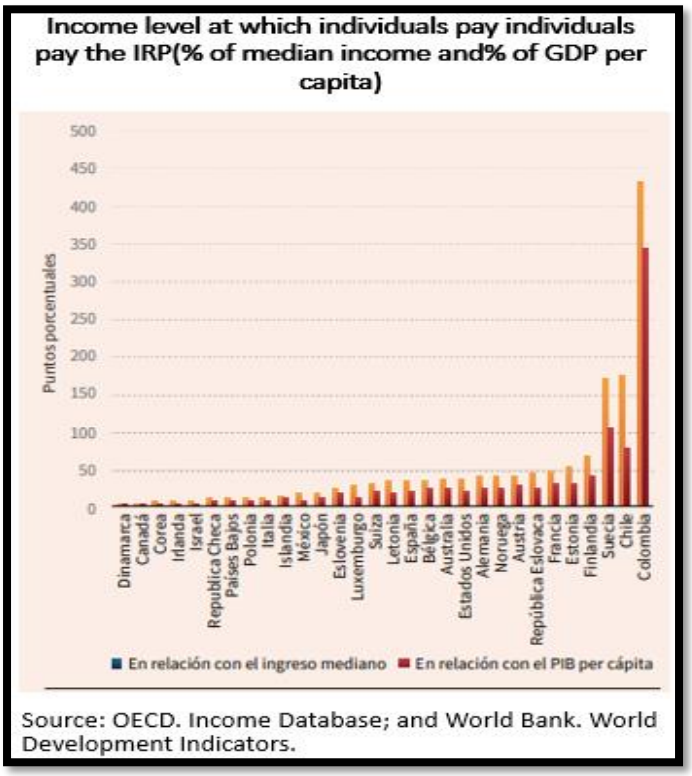


Fig6.Income level at which individuals pay individuals pay the IRP.

In addition, value-added tax (VAT) exemptions and zero-rating, which are intended to make VAT less regressive, end up giving large tax breaks to high-income earners: more than half (57%) of VAT tax expenditures benefit the top three deciles of the income distribution. In addition, cash transfer programs and subsidies for gas, water, and electricity suffer large leakages to high-income households. It is estimated that, according to their socioeconomic profile, more than 65% of households receiving subsidies should receive a lower subsidy or no subsidy at all. Finally, the public pension system generates implicit (and quite

generous) subsidies that accrue mainly to high pension beneficiaries (World Bank, 2021).

the families in action initiative help the poor but a large part of the program reaches the non-poor as evidenced in the following graph.

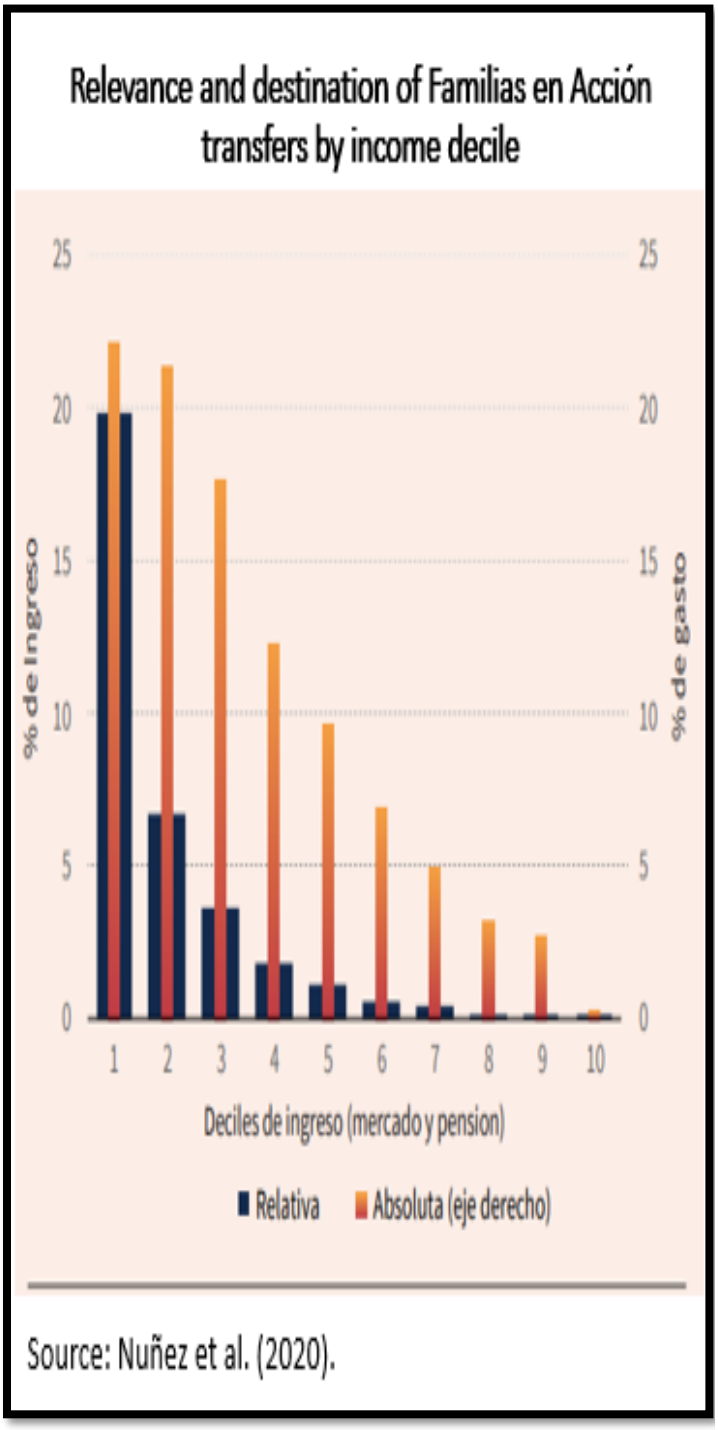


Fig7. Relevance and destination of Familias en Acción transfers by income decile.

Internal and external shocks hinder progress toward equality.

The COVID-19 shock increased poverty by 6.8 percentage points in 2020, and 3.6 million more people became poor, particularly in urban areas. It also caused extreme poverty to increase by 5.5 percentage points, leaving 2.8 million more people unable to meet basic food needs.

The pandemic has also exacerbated inequalities in human capital: the learning poverty rate7 among 10-year-olds is expected to increase from 53% to 60% if schools maintain a hybrid learning program through 2021, or from 63% if distance learning continues year-round. However, COVID-19 is just one of the extreme shocks that can worsen inequalities (World Bank, 2021).

Climate shocks can, too. Estimates indicate that households in the bottom two quintiles of the income distribution would suffer income losses from climate change shocks that, on average, are 1.5 to 1.6 times higher (as a percentage of pre-shock income) than those suffered by the top quintile.

Rural households are expected to suffer income losses that, on average, are 1.8 to 1.9 times higher than those of urban households. However, social assistance programs are not designed to flexibly protect households against shocks, especially weather-related shocks, and the resulting depletion of assets. The lack of a social registry with dynamic and accurate household information, and equipped with climate risk assessment tools, limits the ability of these programs to adapt to new circumstances (World Bank, 2021).

Economic and Social Inequality

Income inequality in Colombia is very high. In 2019, it was the highest among all OECD countries, and most Latin American and Caribbean (LAC) countries. Moreover, inequality in Colombia has been on the rise since 2018 and was further exacerbated by the impact of COVID-19.

At the rate of decline observed between 2008 and 2019, just before the pandemic broke out, it would take Colombia at least three and a half decades to reach the average level of inequality of OECD countries.1 Beyond income inequality, Colombia scores poorly on many other dimensions of inequality (World Bank, 2022).

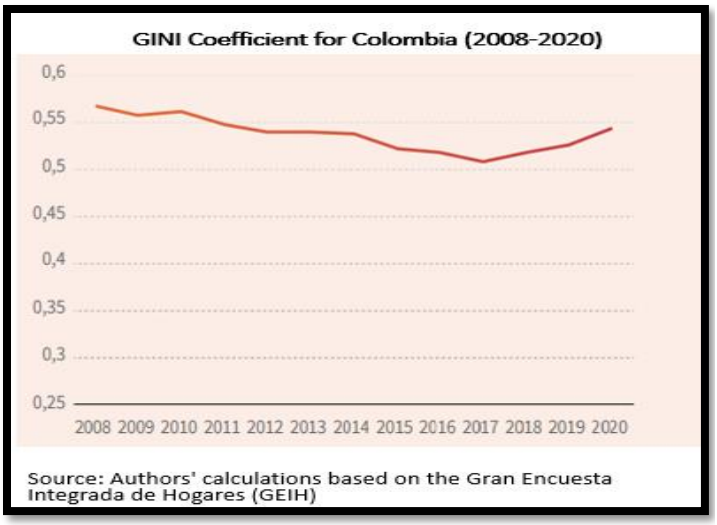


Fig8. GINI Coefficient for Colombia (2008-2020).

The following graphs show the current welfare ratio between Colombia, the OECD and the average of Chile and Mexico, two major references for Latin America.

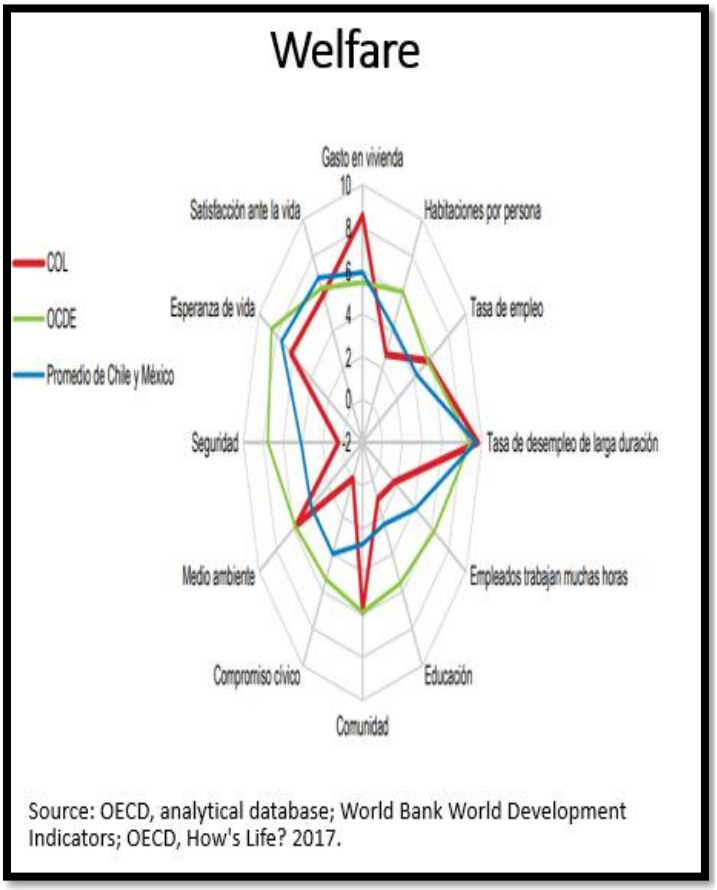


Fig9. Welfare indicators

Figure 9 considers different indicators of well-being. For each indicator, the Figure shows where Colombia ranks, in terms of inequality, relative to the best-performing countries in the OECD and LAC (distance to the frontier). In this international comparison, Colombia ranks at the bottom of many indicators (World Bank, 2022).

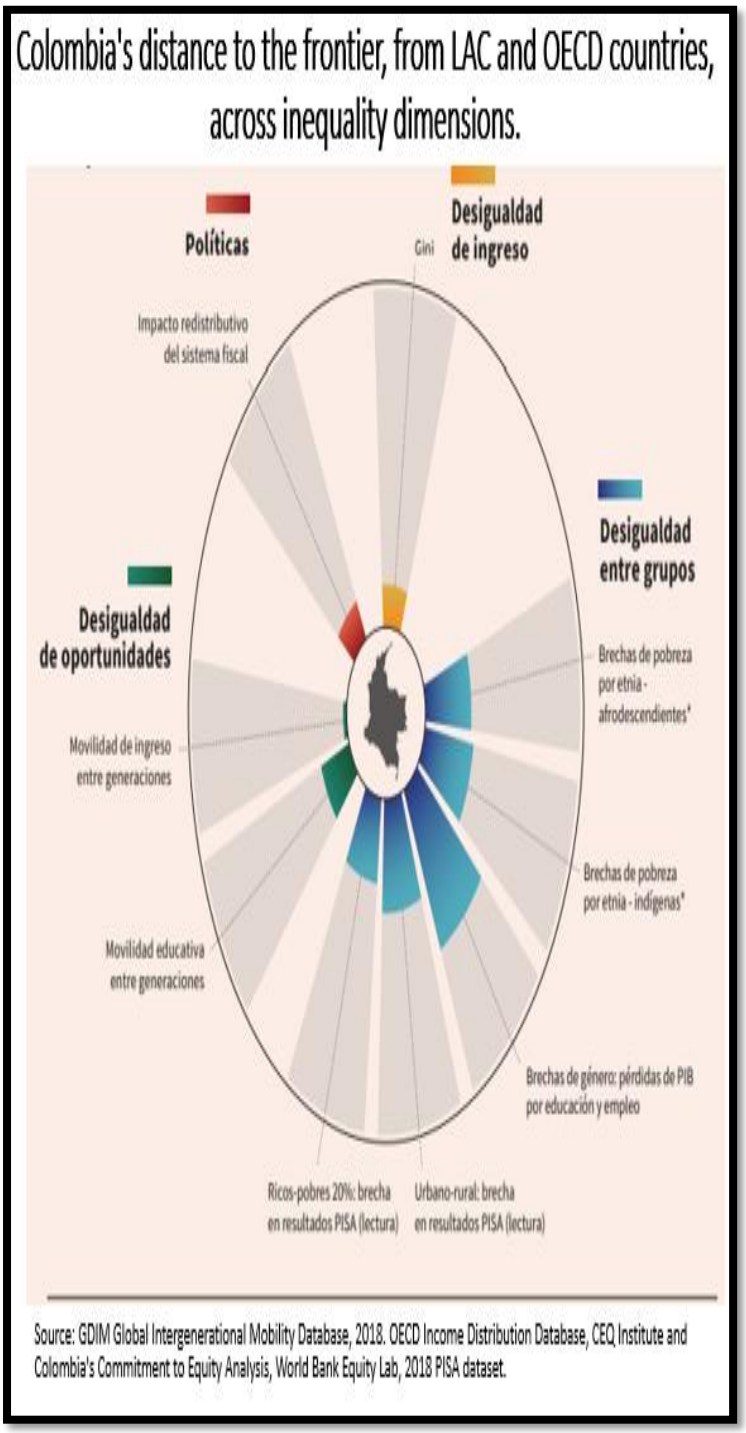


Fig10. Colombia's distance to the frontier, from LAC and OECD countries, across inequality dimensions.

Inequalities are particularly acute among specific population groups, to the detriment of women, people living in rural areas, indigenous groups, Afro-descendants, and migrants (World Bank, 2021).

Poverty rates are significantly higher in rural, migrant, indigenous, and Afro-descendant households (Figure 1.3).2 A Colombian born in Chocó is five times more likely to be born into poverty than one born in Bogotá. Similarly, a woman in Colombia is 1.7 times more likely to be unemployed than a man (World Bank, 2021).

An indigenous Colombian has on average two fewer years of schooling than a non-indigenous one. Two-thirds of the children of Venezuelan migrants (almost 250,000 children, ages 5 to 18) are not enrolled in school, in contrast to other children, who account for 8%. Afro-Colombians are twice as likely to live in slums compared to non-Afro-descendants. When group characteristics overlap, opportunity exclusions are mutually reinforcing. For example, the income gap between the average Colombian woman and man is larger if the woman is indigenous or if she lives in a rural area if other factors do not intervene (World Bank, 2021).

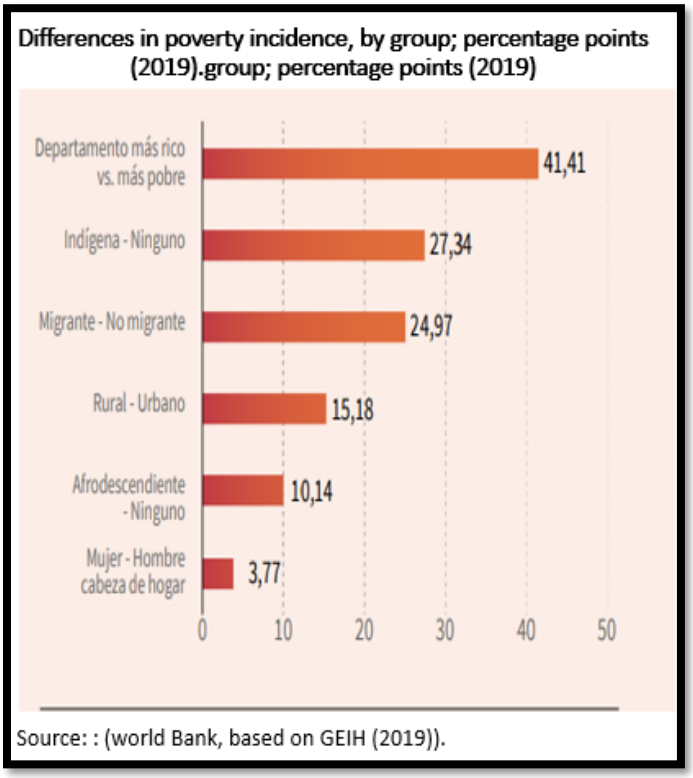


Fig11. Differences in poverty incidence, by group; percentage points (2019) group.

Inequalities also persist over generations, and Colombia has one of the highest rates of persistence of inequality from one generation to the next. Colombia has one of the highest rates of persistence of inequality from one generation to the next. Parents' level of education has a strong positive influence on children's educational attainment.3 This is common in many countries. However, among 146 countries included in the World Bank's Global Database on Intergenerational Mobility (GDIM), Colombia ranks 122nd in the persistence of education across generations and ranks equally low in educational (World Bank, 2021).

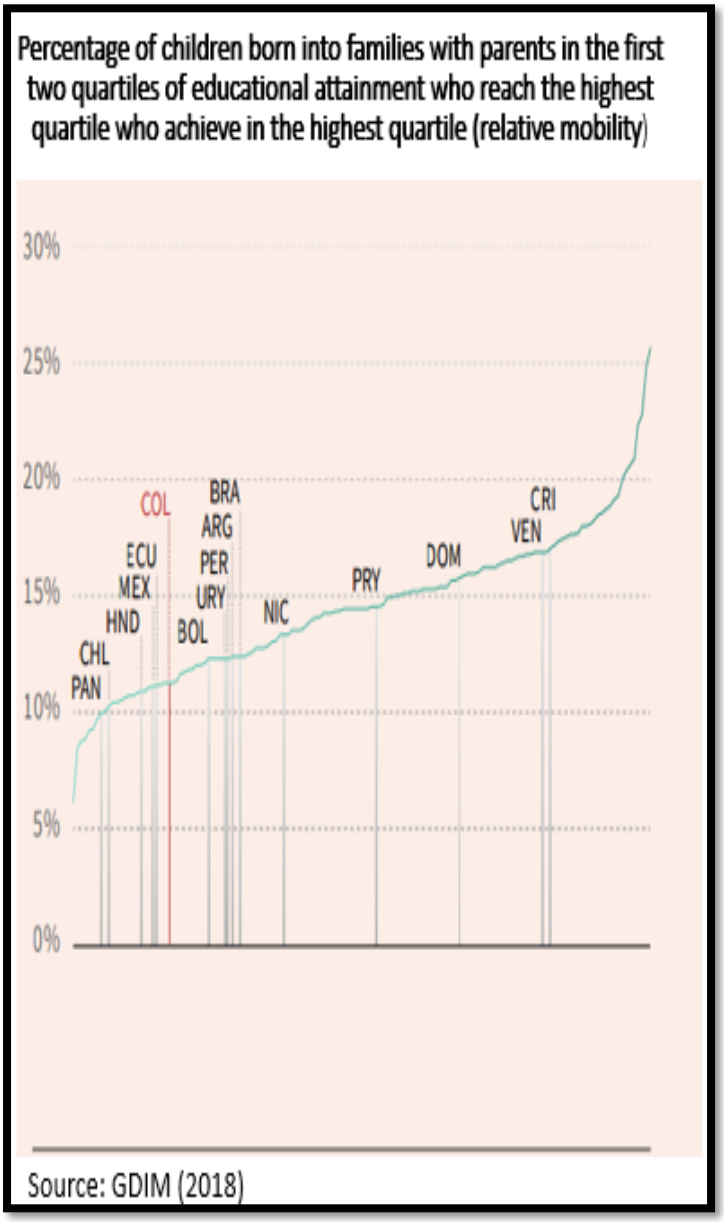


Fig12. Percentage of children born into families with parents in the first two quartiles of educational attainment who reach the highest quartile who achieve in the highest quartile (relative mobility)

This is so, despite advances in educational attainment, even though the proportion of children with a better education than their parents has increased. below is the absolute mobility graph that evidences the percentage of children with a higher educational level than their parents.

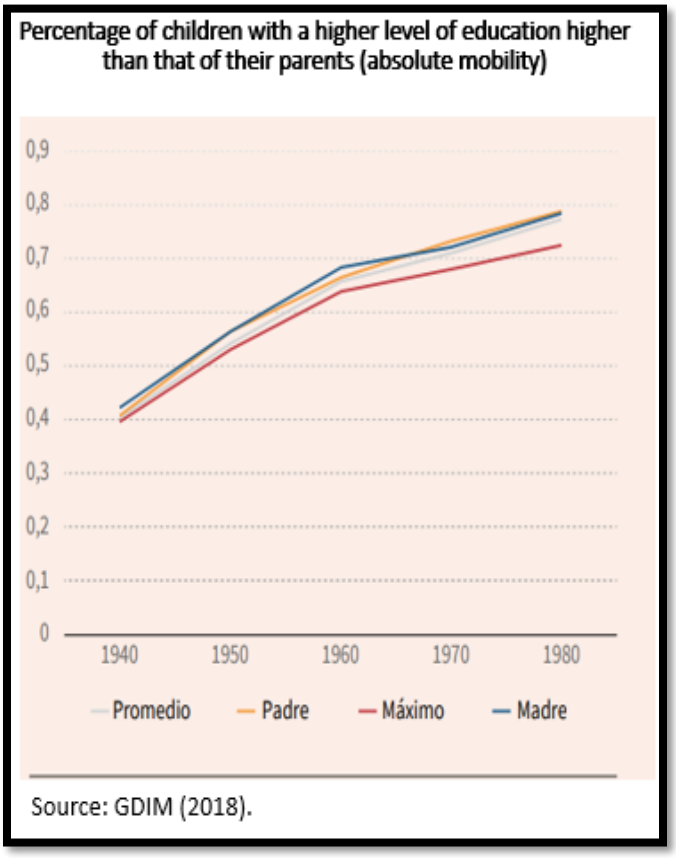


Fig13. Percentage of children with a higher level of education higher than that of their parents (absolute mobility).

In Colombia, if a person is born to parents in the bottom half of the educational attainment scale, his or her probability of reaching the top 25% of educational attainment is only close to 10% (this is around 15% in medium-developed countries) (Narayan et al. 2018). Even more salient is the persistence of income from one generation to the next. Among the 75 countries for which data on intergenerational income persistence are available, Colombia ranks first (World Bank, 2021).

Note: 75 countries are included in the analysis. Intergenerational income mobility should be read as follows: if a Colombian earns twice as much as another man earns, the first man's son can be expected to earn more than twice as much as the first man's son as an adult. the son of the first man can be expected to earn, as an adult, more than twice as much as the son of the lower-earning man (because the of the man with the lowest income (since the coefficient for Colombia is greater than 1), (World Bank, 2018).

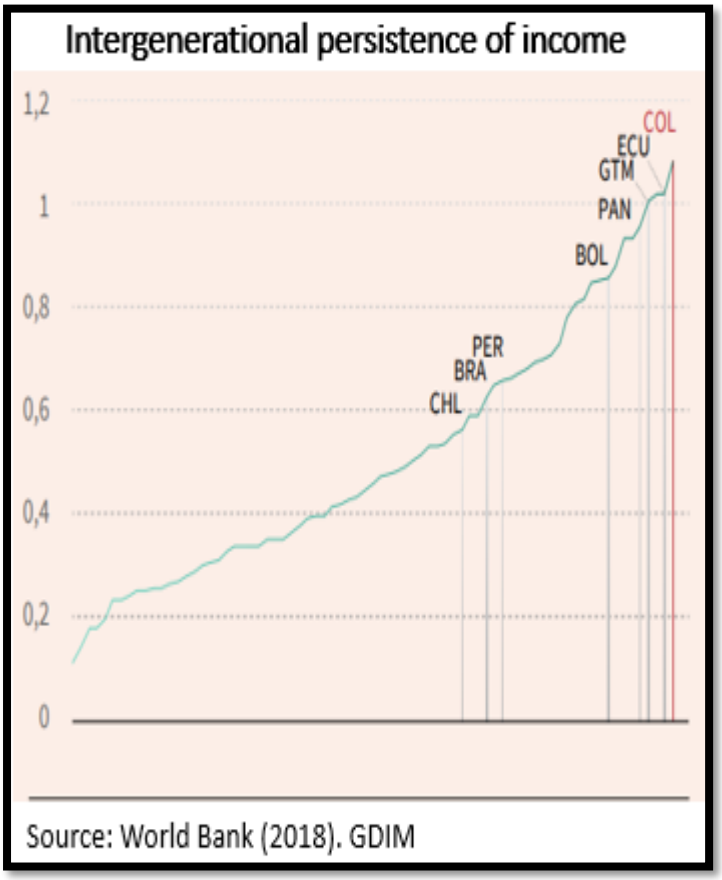


Fig14. Intergenerational persistence of income.

Inequality worries Colombians. Approximately four out of five Colombians believe that income distribution is unfair or severely unfair, and this perception has increased even in periods of slight declines in the Gini coefficient.

At the political level, the deep inequalities that exist in Colombia have not gone unnoticed, and have gained prominence in recent years. Mitigating inequality has been high on the government's agenda, as set out in the National Development Plan (2018-2022). Today, addressing inequality is even more important to the goal of reducing poverty in Colombia, where the COVID-19 crisis has erased more than a decade of progress.

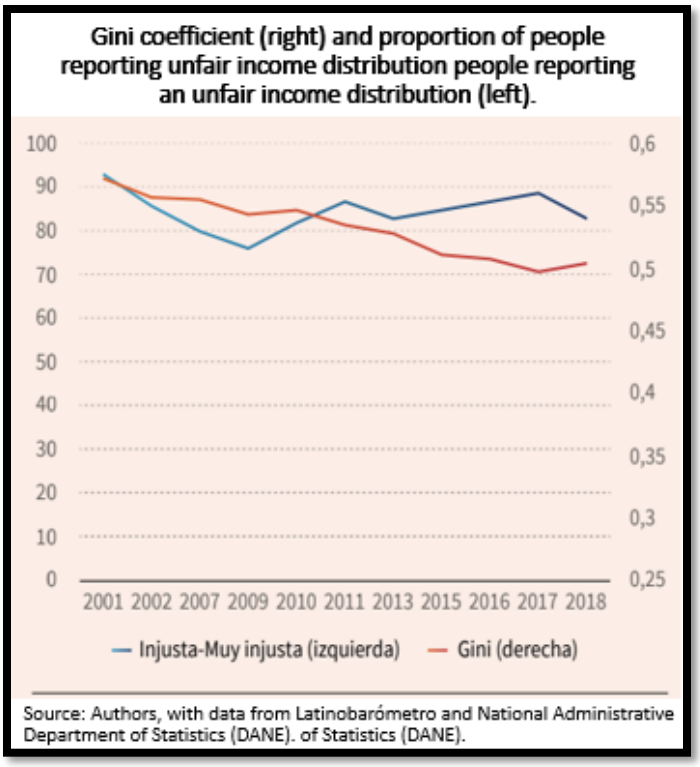


Fig15. Proportion of people reporting unfair income distribution people reporting an unfair income distribution.

Socioeconomic stratification: a public policy of solidarity and redistribution of solidarity and redistribution in the SPD tariff

Under the premise: "Colombia is a Social State of Law founded, among other constitutional principles, on solidarity and the redistribution of the income of the people who make it up", the justification for the creation of stratification in Colombia is generated.

With stratification, the Colombian state has sought to ensure that households with higher incomes economically assist the lower-income population to access, via subsidized rates, household public services (water and sewerage, sanitation, electricity, and natural gas) to achieve universal coverage. In addition to being a mechanism for the allocation of other subsidies and aid to those in the first deciles of payment capacity (differential payments in the education system, differentiation of property tax rates, and differentiation of rates for payments to the State, among others) with time, stratification has become established both in material life and in the collective imagination as an element of social and economic, cultural, and ideological differentiation of the population (Bonilla, López, Sepúlveda, 2014).

The document: “Socio-economic stratification as a targeting instrument” by (Mina,2004) carries out an in-depth analysis of the stratification process in Colombia, starting from the background and foundations for its creation, its implementation, its impact on the population, its successes and failures. Based on this article, the origin and foundations of this concept are presented.

The cadastral appraisal was the basis for the application of public utility tariffs throughout the seventies; since the early eighties, the DANE gave a series of definitions that constituted the starting point for the creation of the six (6) existing socioeconomic strata. During the 1980s and early 1990s, the companies classified the users of public services into the same number of strata but based on their criteria.

In 1992, the National Planning Department was commissioned to design a methodology for urban and rural areas. Law 142 or the Residential Public Services Regime of 1994 created socio-economic stratification as the indicator that governs the tariff policy. Socio-economic stratification has been designed to facilitate the application of differential tariffs to different users of household public services, to help select a certain target population among those with fewer resources, and to focus on some social programs.

According to the National Human Development Program (2003), throughout the 1990s, spending on drinking water and basic sanitation represented between 0.15 and 0.41 percent of the country's Gross Domestic Product. In 2001, the share of such expenditure as a proportion of GDP was 0.20 percent. This percentage, which might seem low, is comparable to that allocated to the group of services that make up social assistance and welfare. What are those provided by the Colombian Institute for Family Welfare (ICBF), and all programs associated with the care of the elderly, children, the disabled, the mentally ill, and other family-related social protection services. The magnitude of spending on public utilities, therefore, merits an evaluation of how users with low capacity to pay are identified and to whom subsidies are allocated.

Together with the SISBEN and the Unsatisfied Basic Needs Index (NBI), socio-economic stratification is one of the main tools for targeting public spending, and it is therefore vital to evaluate its capacity to identify the poorest population.

The socio-economic stratification methodology

The stratification adopted in the different municipalities of the country includes variables related to the characteristics of the dwellings and their environment. A total of eight (8) variables are taken into account:

1. Existence of dwellings on the side of the block with the main entrance.

2. The type of access routes on the street or road on the block side: footpath or road, pedestrian, vehicular or pathway, pedestrian, vehicular on dirt, vehicular on gravel, vehicular on concrete, asphalt or asphalt vehicular on concrete, asphalt, or cobblestone.

3. Presence of pollution sources on the side of the block or in front of the block: sewage in view of the block. sewage in sight, rubbish dumps, slaughterhouses, market place, workshops, factories, terminals, etc. marketplace, workshops, factories, bus terminals, canteens, bars, etc.

4. Predominance of pavements on the side of the housing block without sidewalks, with sidewalks but without pavements, with pavements but without green space, or with both. The predominance of dwellings on the block side without a front garden, with either a small, medium, or large front garden.

6. Presence of dwellings without garages on the block side or with garages with different characteristics.

7. Material of the facades on the block side: Guadua, cane, matting, board, or waste; uncovered, i.e. adobe, wattle, and daub, brick, brickwork, prefabricated slab, block, or brick; in plaster, rendering, or rendering (unpainted or painted; with veneer, in polished brick or in fine wood. or in fine wood.

8. Type of material of the main door: board, bamboo, matting, zinc or fabric; polished wood, sheet metal, wood veneer, wood veneer. polished wood, sheet metal, worked iron or aluminum frame; finely carved or completely aluminum; finely carved wood or entirely in glass.

Although the methodology has as its central axis the eight variables noted above, there are some differences between the different cities, which vary according to the number of inhabitants and blocks, the number and type of economic activities carried out in them, and the degree of Unsatisfied Basic Needs (UBN). The housing is classified in one or another of the six (6) strata adopted in most municipalities in the country.

The socio-economic strata in which the dwellings and/or properties can be classified are 6, denominated as follows:

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

Table1: Socio-economic stratification in Colombia.

Source: Own elaboration

"Of these, strata 1, 2, and 3 correspond to low strata that house users with lower resources, who are beneficiaries of subsidies for residential public utilities; strata 5 and 6 correspond to high strata that house users with greater economic resources, who must pay additional costs (contribution) on the value of residential public utilities. Stratum 4 is not a beneficiary of subsidies, nor does it have to pay cost overruns; it pays exactly the value defined by the company as the cost of providing the service. The classification in any of the six strata is an approximation to the hierarchical socioeconomic difference, read poverty to wealth or vice versa" (DANE, 2023).

Thus, the tariff not only determines to a large extent the value be paid, since its components generate considerable differences between the tariffs managed by the different companies that provide the same service, but it does not depend on the users' ability to pay (socio-economic condition of those who demand them) and does not even take it into account, as occurs when estimating the price of most goods and services. The subsidy or contribution percentage (P%) consists of optional caps or maximums, theoretically constant at the national level for each socio-economic stratum, as established by law (Table 2) (CELAC, 2006).

Strata and subsidy or contribution factor

Estrato	Porcentaje de subsidios (Factor)
1: bajo-bajo	0,50 (subsidios hasta 50%)
2: bajo	0,60 (subsidios hasta 40%)
3: medio-bajo	0,85 (subsidios hasta 15%)
4: medio	1,00 (sin subsidio ni contribución)
5: medio-alto	1,20 (contribuciones hasta 20%)
6: alto	1,20 (contribuciones hasta 20%)

Law 142 of 1994 Colombia

Table2. Strata and Subsidy or contribution factor

The legislature, in approving the SPD regime, thus defined the extent to which socio-economic conditions can affect billing: no matter how poor a residential user is, he will only be eligible for SPD if he pays 50% of what it costs to provide it, even if the bill could be considered onerous; likewise, no matter how rich a user is and even if the tariff is very low, he will only be obliged to contribute 20% more than what it costs (CELAC, 2006).

On the other hand, non-residential users, i.e. commercial, industrial, and institutional users, also contribute to subsidizing the lower strata. Exempt from this contribution, at the customer's request, are hospitals, clinics, health posts and centers, and non-profit educational and welfare centers.

Article 2 of Law 632 of 2000 provided that the surcharge factor should be adjusted to the percentage necessary to ensure that the amount of contributions is sufficient to cover the subsidies applied (CELAC, 2006).

The stratum to which the aforementioned subsidy or contribution percentage corresponds, as indicated, is the one resulting from "the classification of residential properties in a municipality", maximum in six groups, implicitly considered a proxy of economic capacity, and in accordance with the levels that have distinguished the tariff system for several decades. The unit value (reference cost or tariff by subsidy or contribution percentages, by service and billing period according to the company) is, then, the one that is affected by the socio-economic stratum.

It is not the tariff itself, as has commonly been suggested. In this way, a residential user, depending on the stratum in which his or her dwelling is classified, is billed for consumption at a certain percentage of the tariff (not the total consumption but the basic or minimum subsistence consumption,10 or subsidized consumption in the case of strata 1, 2 or 3) (CELAC, 2006).

"As a result of this classification in the same city, one can find homes as dissimilar as those ranging from the slum that expresses -without a doubt- the misery of its inhabitants, to the mansion or palace that, in the same way, evidences an enormous accumulation of wealth. The same happens in the rural area with houses ranging from shacks without walls to "ranches", haciendas with large extensions of productive land, and recreational estates with exuberant comforts" (DANE, 2023).

In addition to being a mechanism for the allocation of other subsidies and aid to those in the first deciles of ability to pay (differential payments in the education system, differentiation of property tax rates, differentiation of rates for payments to the state, among others), over time, stratification has become established both in material life and in the collective imagination as an element of social and economic, cultural and ideological differentiation of the population (CELAC, 2006).

Socio-economic stratification seeks to respond to the purpose of identifying subsidized or subsidized housing, based on two basic assumptions. On the one hand, it is possible to discriminate against users according to their income level, payment capacity, or socioeconomic characteristics.

On the other hand, the characteristics of the environment make it possible to define homogeneous groups of users for the differential collection of SPD. The assumption of the existence of these relationships led to a stratification methodology that does not explicitly consider aspects associated with the socio-economic characteristics of households, but rather classifies them by seeking homogeneities directly on the characteristics of the properties and their surroundings (Bonilla, López, Sepúlveda, 2014)

Socio-economic stratification and household income and expenditures

Income is not among the variables that are part of socio-economic stratification. It is based, as noted above, on the characteristics of the dwelling and the conditions of the environment in which it is located. However, for equity purposes, it is expected that there is a high correlation between the distribution of households resulting from applying the stratification method and that obtained from income. This should be the case since one of the objectives of stratification, and perhaps the most important one is to cross-subsidize, i.e. to redistribute income (Mina, 2004).

Table3 presents the distribution of households according to their share of total income resulting from stratification. In terms of income, the results do not indicate a very equitable situation. Households at all income levels are part of stratum 1: about 46% of this is made up of households in deciles 4 to 7, and only 31% of other households in the first two deciles. Meanwhile, households in stratum 2 are made up of about 50% of people from deciles 5 to 9. Although 71% of people in stratum 6 are from the top decile, there are people in the bottom five income deciles (8.2%). Given the above, payments for residential public services that adopt the stratification methodology do not seem to be in accordance with the users' ability to pay, understood as the household's income level (Mina, 2004).

Distribution of households by income decile and socioeconomic stratum (2003)

Decil de Ingreso	Estrato Socioeconómico para Pago Energía Eléctrica						Total
	1	2	3	4	5	6	
1	13,6	10,8	5,2	1,1	1,0	0,7	683.744
2	17,2	11,6	6,0	1,5	0,6	2,6	782.085
3	13,8	12,4	7,1	2,5	0,4	3,9	803.178
4	13,6	13,4	9,8	1,9	2,7	0,3	905.529
5	14,5	12,8	10,8	5,3	2,2	0,8	945.864
6	10,6	11,4	12,6	5,7	4,1	2,6	911.338
7	7,0	11,5	14,1	8,7	5,1	6,1	939.923
8	5,4	8,8	14,6	15,6	10,5	3,5	898.155
9	2,2	5,6	12,1	24,0	19,0	9,1	757.447
10	1,9	1,8	7,7	33,7	54,3	70,6	710.196
Total	1.232.679,0	3.326.332,0	2.810.317,0	644.929,0	196.722,0	126.480,0	8.337.459

Socioeconomic stratification as a targeting instrument, Mina(2014)

Table3. Distribution of households by income decile and socioeconomic stratum (2003).

In other words, as of 2003, the conclusion remains the same as VÉLEZ (1996, p. 216): The process of socioeconomic stratification fails to efficiently distinguish the richest households from the middle-income and poorest households.

As a result, there is a high proportion of high-income household's subject to tariffs and subsidies very close to those corresponding ideally to the lower strata. According to VÉLEZ (1996), in an ideal stratification, the first stratum should only include households in the first two income deciles, the second stratum is made up of households between the second and fourth income deciles, the third stratum includes households between the fourth and sixth, and the sixth stratum would include households in the top two income deciles.

So far, a fairly low correlation has been found between stratification and income, which, measured through Pearson's coefficient, although significant at a level of 1%, only reaches a value of 0.249.

The problems of socioeconomic stratification. A gradual erosion of the instrument

Several studies on the evaluation of stratification and its possible relationship between household economic capacity and the characteristics of households using public services highlight problematic aspects of the redistribution instrument. The discussion of how to approach the socioeconomic conditions of the inhabitants of stratified households has been the subject of different studies and proposals. Some attempt to establish the correlation between income, expenditure levels, and housing characteristics. Others correlate the economic capacity of households, measured from the quality of life indicators or poverty levels, with the characterization of the dwellings they inhabit.

Still, others connect the characteristics of homes and environments based on multidimensional perspectives of human well-being. More recently, other approaches correlate the economic capacity of households with the quality of urban life enjoyed by users of public services derived both from their economic level and the environment and the quality of public goods offered by cities. In all cases, the approximations coincide with the fiscal policy perspective that assumes that the principles of benefit and ability to pay to refer to divergent conceptions of justice to solve the social problem of

financing public goods (Bonilla, López, Sepúlveda, 2014).

Based on the document: "La estratificación socioeconómica para el cobro de los servicios públicos domiciliarios en Colombia ¿Solidaridad o focalización?" prepared by the United Nations - ECLAC, 2006. The following findings are described:

The Mission to Support Decentralization and Targeting of Social Services or Social Mission of the DNP, in 1993, contracted the study Social Expenditure in SPD of the aqueduct, sewerage, electricity, and gas, carried out by the firm Econometría S.A., under the socioeconomic stratification for the collection of domiciliary public services in Colombia Solidarity or targeting? directed by Álvaro Reyes Posada and coordinated by Carlos Eduardo Vélez, and Ernesto May and Ariel Fiszbein on behalf of the World Bank, the results of which were succinctly published in 1995.26

In his study, Vélez evaluates the discriminatory power of socioeconomic stratification using per capita income as an indicator of the economic status of each household. To this end, he measures the statistical correlation between the socioeconomic stratum of each household and its respective income decile, assuming that the stratification should agglutinate the greatest number of users in the ideal stratification band, which he illustrates as follows:

Socioeconomic strata and income deciles: "ideal" stratification and type i and ii errors.

Decil	Estrato					
	1	2	3	4	5	6
1						
2	ESTRATIFICACIÓN			*Error de*		
3				*tipo I*		
4		"IDEAL"				
5						
6			ESTRATIFICACIÓN			
7		Error de				
8		*tipo II*		"IDEAL"		
9						
10						

Source: C. Vélez, 1995.

Type I error: classified in a stratum higher than the one corresponding to their income level.

Type II error: classified in a lower stratum than the one corresponding to their income level.

Table4. Socioeconomic strata and income deciles: "ideal" stratification and type i and ii errors.

The researcher finds a "weak association" between housing quality (wealth) and the strata, which he attributes to possible technical deficiencies in the qualification system or to an interesting manipulation of it. Vélez (1995) goes on to note that the evaluation of stratification; shows that it functions precariously in the eight main Colombian cities investigated in the 1992 National Household Survey No. 77, the source of information for his study.

The quality of the stratification, calibrated against the income deciles of this group of cities, leaves much to be desired since the Pearson correlation coefficients are less than half the feasible level under an ideal stratification.

Thus, for 38% of the SPD user households in these cities, type II stratification error is committed, that is, more or less two out of every five households are classified in strata lower than those that would ideally correspond to them according to their income level; and almost a quarter of the user households are subject to type I error, that is, they are classified in strata higher than those that correspond to them according to their income level.

The inefficiency of stratification, according to the study, is more marked in Bucaramanga, Cartagena, Manizales, and Pasto than in large cities (United Nations - ECLAC, 2006).

Likewise, the study by Libia Martínez (2004) wonders about the validity of the variables of the DNP stratification model to achieve an adequate classification of people in terms of their income, and states that stratification is not an indicator of the quality of life, inequality or poverty, but strictly speaking it is an indicator of income, concluding that: "The model leaves out many of the fundamental socioeconomic variables so that it can hardly be accepted as reflecting the level of income and its differences and, therefore, inequality..."

She goes on to say that, in terms of informative relevance, it can be affirmed that the indicator is simplistic and does not respond to plural informative bases, since it does not consider variables that identify well-being in a broad sense, such as human capital, people's freedoms, people's perceptions of their living conditions, or variables such as access to SPD, levels of education or access to health services, including in other traditional indicators; and that it is conceived more from the point of view of the spatial distribution of the city's inhabitants, a

necessary aspect for urban planning, but which does not consider the variables mentioned above (2004: 77-78). Incompatibility between DANE's stratified sampling frame and the "sampling frame" of the stratifications made by the municipalities with DNP's methodologies (United Nations - ECLAC, 2006).

The sampling frame is the inventory of all the existing blocks or properties in the urban or rural area, obtained from the national or city censuses. DANE stratified the blocks of the sample frame of 53 cities, starting from updating the cartography or plans that graphically express the spatial location of the dwellings when it carried out the Population Studies between 1980 and 1983. For this purpose, it applied a methodology designed by the entity, consisting of the definition of six strata that, perceptively, the interviewer assigned to each block, without using weightings of the variables involved or a statistical method of classification into strata (See Annex 2). With this same methodology, he stratified only the new developments (blocks) occurring in 1989 and 1993 (United Nations - ECLAC, 2006).

The stratification of the sampling frame "aims to reduce sampling error by seeking the maximum possible homogeneity within each stratum, and the greatest possible heterogeneity between strata" (DANE, 1989); also, there are several reasons for stratifying the frames: in order to have estimated by stratum, to establish the Household and Quality of Life Surveys, it does not use the sample frame of stratified blocks but the sample frame of stratified sections, which is built from estimating the modal stratum of the section (group of approximately 20 contiguous blocks).

In compliance with the legal mandate to use the SPD stratifications to stratify the sample frames (article 13 of Law 505 of 1999), in 2001 DANE informed the DNP that it had done an exercise for the 14 main cities and that it encountered difficulties in assigning it because the blocks did not coincide, The DANE reported to the DNP in 2001 that it had done an exercise for the 14 main cities, and that it had encountered difficulties in assigning it because the blocks did not coincide, in a percentage that ranged between 11% and 26% (except Cartagena, 3%), and because of considerable differences in the proportions of strata, mainly 1 and 2, which is why it had only adopted the SPD strata in the cities of Bucaramanga, Medellín, Cali and Barranquilla (DANE, 2001) (DANE, 2001).

Analyzing the effect of the fact that the information observed (collected) in one stratum is imputed to another stratum, as DANE does, for example, in the ECH (Encuesta Continua de Hogares, formerly Encuesta Nacional de Hogares), is of the greatest statistical importance.

It is equally important to understand that, to the extent that the strata of the DANE sample frames are not the same as the "sample frames or censuses" of the SPD stratification, it is inappropriate to equate both stratifications to examine the targeting or allocation of SPD subsidies, 35 since these subsidies have been granted to households other than those eligible for subsidies as reported by DANE, which in the past were those of the stratification of each company, and today are the result of the stratifications of the municipalities-DNP-companies (United Nations - ECLAC, 2006).

The inappropriateness of comparing the LCS strata of different years and asserting that the changes are due to discretional stratification management to explain the changes that occurred in the stratification of the country during a period, and even more delicate, to analyze the targeting of SPD subsidies, the researchers of the studies reviewed used the stratifications of the DANE LCS.

These surveys are not statistically representative by strata; nor do they investigate the same units: cities or municipal capitals, rural population centers, and rural areas with scattered houses and farms; and they do not even allow comparison between regions, since some are different (Bogotá was with Soacha in the 1997 survey, the Pacific included the Valley, and the Atlantic included the Archipelago).

As can be seen in the first table of the previous section, the distributions of strata vary significantly during the period, without the researchers who have used them, nor DANE itself, providing explanations for data that are taken for granted. If we do not take into account that the stratifications registered by DANE in the three surveys, as indicated above, correspond to stratifications from different sources, it can be lightly asserted, as almost all of the researchers cited above do, that the changes are due to excessive discretion in the administration of the system or manipulation of the results by the mayor's offices, even when they had no authority over the stratifications used to assign SPD subsidies, since in 1993 the stratification was the responsibility of the companies and only became the

responsibility of the mayor's offices, in practice, after 1995, and even later, given the slowness with which they have adopted the studies derived from the DNP's methodologies(United Nations - ECLAC, 2006).

Other studies warn about the presentation of critical errors in the stratification process, such as the case of the book Socio-economic stratification and cadastral information. Introduction to the problem and future perspectives by researchers BONILLA, J., LÓPEZ, D., and SEPÚLVEDA RICO, C.E., LÓPEZ CAMACHO, D., and GALLEGO ACEVEDO, J.M., Editorial Universidad del Rosario: Alcaldía Mayor de Bogotá D.C., 2014. In which the problem of stratification is described when studying its relationship with household income is reinforced by taking other indicators of poverty or ability to pay as a reference.

Martinez (2004) study for Bogotá the relationship of strata with the incidence of poverty by unsatisfied basic needs (nbi), the poverty line (lp), and households receiving less than one minimum wage per capita.

The study also shows a correlation with household consumption structure. For all indicators, the existence of inclusion and exclusion errors is clear. Strata that receive subsidies (1, 2, and 3) have a significant percentage of non-poor. If we take, for example, stratum 2, 27% of households in this group are not poor according to the lp, and 22% according to the minimum wage criterion, and yet they receive subsidies for the payment of spd tariffs. Meanwhile, the analysis of current spending is used to prove that spending on food and public services is proportionally higher in the lower strata than in the higher strata, which suggests a regressive scheme in the collection of tariffs.

Meléndez (2004, 2008) proposes a subsidy targeting index that also captures the existence of inclusion and exclusion errors. For the case of water and sewerage in Bogota, it is concluded that by 2003, the system was providing subsidies to more than 80% of non-poor connected households. In general terms, the study states that "although stratification has made it possible to allocate subsidies to poor households, this has been achieved at the cost of huge inclusion errors that translate into wasted resources" (Bonilla, López, Sepúlveda, 2014).

In an evaluation by the National Planning Department (DNP) of stratification as an instrument for classifying users, Econometrica (2006) suggests the need to explore a much more precise definition of capacity to pay.

Based on a Paretian conceptualization (consumer theory) proposed to refer to minimum or subsistence levels in the expenditure of goods and the problem of their optimization, a set of variables that make up a more elaborate indicator of the ability to pay is analyzed to propose a comparison between initial indicators of permanent disposable income and recurrent expenditure (which includes expenditure on food and housing) with data from the co-censal survey of the 2005 Census. From this, the correlation between the ability to pay and socioeconomic and poverty variables is established, on the one hand; and the explanation that the ability to pay can offer with respect to stratification is sought, on the other (Bonilla, López, Sepúlveda, 2014).

Based on a review of the most problematic aspects involved in stratification in the diagnoses carried out at different times, studies on the application of the instrument show that, over the years, it ceased to reflect the unequal economic capacity approximated from the different approaches (Bonilla, López, Sepúlveda, 2014).

Consequently, exclusion and inclusion errors increased, calling into question the efficiency and effectiveness of the stratification mechanism. The basic information of the model had lost validity and had even been manipulated by political interests (authorities assigning the stratum according to electoral interests, for example), or simply had not been flexible enough to respond to changes in the urban structure (Bonilla, López, Sepúlveda, 2014).

At the beginning of the 21st century, it was clear to the country that the stratification of the previous ten years was inefficient due to errors of inclusion and exclusion, lack of monitoring of socio-economic dynamics, and the use of the information that fed the classification model. This was recognized in the Conpes 3386 document of 2005 (Bonilla, López, Sepúlveda, 2014).

System for the Identification of Potential Beneficiaries of Social Programs: Sisbén

Based on the web portal https://www.sdp.gov.co/gestion-estudios-estrategicos/sisben/preguntas-frecuentes corresponding to the Secretary of Planning of the Mayor's Office of Bogota, Sisbén and its main characteristics are defined as Identification System of Potential Beneficiaries of Social Programs, which allows classifying the population according to their living conditions and income.

This classification is used to focus on social investment and ensure that it is allocated to those who need it most. The Sisbén is an information system that identifies and classifies the population according to its socioeconomic and social situation through a survey that ranks the population according to its social and economic status.

The Sisbén information is used by social programs to identify their beneficiaries; being in the Sisbén does not guarantee automatic access to social programs or benefits.

The entities in charge of the social programs are the ones that define the cut-off group and requirements to link their beneficiaries. From 1995 to date, the Colombian State has implemented four versions of the Sisbén. Each version has a methodology, i.e., a set of procedures to obtain the result that reflects the socioeconomic conditions of each person. The fourth version - Sisbén IV was implemented on March 5, 2021.

In Sisbén IV there are four groups that are denominated by letters. Each group is composed of subgroups identified by a letter and a number that allow for a more detailed classification of individuals.

There are four groups in Sisbén IV: group A, made up of the population with the lowest income-generating capacity or population in extreme poverty; group B, made up of poor households, but with greater income-generating capacity than those in group A; group C, made up of the population at risk of falling into poverty (vulnerable); and group D, made up of the non-poor population.

Classification	Conformation
Group A	Made up of 5 subgroups (from A1 to A5)
Group B	Made up of 7 subgroups (from B1 to B7)
Group C	Made up of 18 subgroups (from C1 to C18)
Group D	Made up of 21 subgroups (from D1 to D21)

Table5: Classification and conformation of the groups of Sisbén IV

Source: Own elaboration

Who participates in the Sisbén?

I. National Planning Department: Develops the Sisbén Methodology; designs the socioeconomic characterization form and the technological tools with which the surveys that feed the Sisbén database are applied; Applies quality controls, and validates the consistency of the information of the surveys and updates received from the Municipalities. Validates and publishes the certified information on the web page www.sisben.gov.co.

II. Territorial Entities - District Planning Secretariat: Implements administer and operate the Sisbén in Bogotá D.C. Applies the surveys following the guidelines established in the Methodology, sends the information to the National Planning Department for processing, validation of information consistency, certification, and publication of the classification result in www.sisben.gov.co, registers the information update procedures requested by citizens with which the Sisbén database is updated - sends the information of the update news to the National Planning Department for validation, certification, and publication within six working days after being received.

III. The entities that administer the social programs: Establish the cut-off group and requirements for admission to the programs. They select the potential beneficiaries of the social programs that meet the requirements.

IV. Citizens: Provide truthful information, under oath for the completion of the survey must keep updated the information of your household registered in the database.

Difficulties presented by Sisbén

The Sisbén has received strong criticism in its implementation, including its foundations and creation, beyond the already complexity corresponding to the selected tool that constitutes an on-site survey that any Colombian citizen can request, the socio-economic stratum of the dwelling is one of the variables of the Sisbén survey and this data is taken from what is registered in a domiciliary public service receipt that the household must present at the time of the survey (DPN, 2021), The aim is that those who are registered in the lower strata (1,2,3), ideally constituted by the most vulnerable population, will have access to the different subsidies and social programs that allow them to mitigate their poverty indexes and access public policies such as education subsidies that allow them to generate social mobility; however, the strong shortcomings of stratification in Colombia as a targeting instrument have been widely presented in the previous sections of this theoretical framework.

Through the article "Identification and affiliation of beneficiaries - Sisbén" developed by researchers Oscar Fresneda and Patricia Martínez for the Universidad Nacional, critical points are established in the selected methodology as well as difficulties in the implementation process of the surveys that generate the Sisbén stratification, the most relevant of which will be presented below.

The survey conducted in the municipalities also questioned municipal officials about problems that in their opinion have affected the quality of the Sisbén. Among those considered were circumstances of a different nature. The problem reported by most municipalities is the lack of technical standards in the application of Sisbén (50%), and once again, budgetary limitations (47.5%). They are followed by a lack of techniques in the application and updating of databases, as well as a lack of instruments to identify the poor, which are reported by 35.5%, 33.8%, and 20.1% of the municipalities respectively. These results indicate a perception of the absence of regulations for the operation of the Sisbén, which technically affects its administration. 22% of the municipalities have a perception of political management in the Sisbén that affects the implementation of the Sisbén and 12.3% are not affected by any of the mentioned problems.

The lack of technical standards is a problem with more frequent expressions in the medium and small municipalities, which double the relative frequency of the large ones. And, in contrast, it is in the large municipalities where there are greater difficulties in updating the databases. Political management interference also has a higher proportion in large municipalities: 44.8%. In the medium-sized ones it reaches 40.7% and in the small ones 16.6%.

This situation corresponds to that reported on arbitrary entries and exits from Sisbén. In 39% of municipalities, it is reported that this situation has occurred in Sisbén levels 1 and 2. These are phenomena that affect the quality and bias of the beneficiary selection and affiliation processes, both due to the influence of client-list practices and other types of interests (economic, family, etc.). It is also in the larger municipalities where it is most prevalent (66%). It occurs in 41.4% of the medium-sized ones and 36.7% of the small ones (graph 34). In the survey of the departments, 64% of the sectional directorates surveyed reported that this phenomenon occurs in the municipalities under their jurisdiction.

From the responses obtained in the structured interview with the sample of municipal administrations, the perception of possible measures to improve the functioning of Sisbén is obtained (Graphs 35 to 38). Of the items considered, the one with the highest frequency was financial support. Of the items considered, the most frequent was financial support. 75.7% of the officials in the municipalities consider this to be necessary assistance. In the larger municipalities, the proportion reaches 93%. National and departmental technical assistance is considered necessary in 59.6% of the municipalities, and it is also in the large municipalities where the highest frequency is recorded: 68.2%. Technical training is perceived as necessary in 55% of the municipalities, and in this case, in the medium-sized ones, there is greater

acceptance: 70.3%. Regulatory development is perceived as necessary in minority proportions: 20.1% of the municipalities; 42.7% in the large ones.

Some features of the situation of the municipalities with respect to the Sisbén administrator are as follows: Almost all municipalities have a Sisbén administrator. Most of those that reported filled this position between 1995 and 1996. Between 1998 and 2000, 22% of the municipalities made their first appointment to the position.

Stability in the position, as an expression of continuity in Sisbén administration jobs, shows high mobility. Between 1998 and June 2000, 52.4% of the municipalities had more than one administrator, and 30.3% had more than two. In 72.9% of the municipalities, the Sisbén administrator is a municipal official. In 60% of the municipalities, the administrator has an exclusive dedication, without sharing other functions. In the medium-sized municipalities, 79% have this dedication and in the small municipalities 45%.

The educational level of the administrator with the highest frequency is high school: 38.3%. 20.6% have completed university studies and 14.6% have completed studies at this level without completing them.

Some 4.3% have a specialization and 21.7% have done technical studies. In the large municipalities, 32% have completed university studies, 23.3% have postgraduate studies or specialization and 40.6% have technical studies. Among the small municipalities, 46.5% have a high school education, 16.6% have incomplete university studies and the same proportion has technical studies.

Twenty-three percent of Sisbén administrators have not received specialized training for the functions of this position. The percentage is 19% in large companies and 23% in small ones. 40.1% of the administrators have received training from the department, 40.7% from the Social Mission, 30.2% from the previous administrator, and 13.6% from individuals.

An expression of the degree of institutionalization of Sisbén is the existence of a working group for its administration. In 65.3% of the municipalities, there is such a group. In 40% of the small municipalities, it has not been implemented, the same happens in 193% of the medium-sized ones and 4.2% of the large ones.

The Sisbén technical committees, contemplated in the technical guidelines given since the beginning of the program, are not constituted in 73.3% of the municipalities; in the large municipalities 31.8% are in this situation, in the medium-sized ones 70.3% and in the small ones 76.7%. Community organization committees on Sisbén have been created in 53.5% of the municipalities. And in this case, they operate more frequently in the small (56.8%), than in the medium (42.1%) and large (40.3%).

Seventy-eight percent of the municipalities report that the main reason for not enrolling those on the waiting list is a lack of budget. This situation affects 80% of the medium and small municipalities and 47% of the large ones. Non-authorization by the department affects 18.2% of the municipalities and has an incidence of 36% in the large municipalities.

Some indications regarding the perception of the adequacy of Sisbén to select the poor were captured in additional questions of the survey: 59% of the municipalities answered that Sisbén does not adequately identify the people who deserve to be in the subsidized regime, 68% in the large municipalities, 70% in the medium-sized municipalities and 57% in the small municipalities. Among the departmental health secretariats, 40% considered that Sisbén adequately selects the people who deserve to be in the Subsidized Health Regime.

Despite the four attempts to generate the Sisbén, Colombia continues to present great deficiencies in the establishment of effective mechanisms to guide public policies in favor of the most vulnerable communities. This is evidenced by the citizen monitor against corruption, which is an observatory of transparency for Colombia (national chapter of Transparency International), which continuously monitors the facts and risks of corruption, as well as public anti-corruption action in Colombia. Through its platform for information curation, research, and monitoring of corruption, it offers citizens open data, study methodologies, and descriptive and analytical reports on this phenomenon, and has found the following findings concerning the Sisbén:

“The System for the Identification of Potential Beneficiaries of Social Programs -SISBEN- is designed to identify and classify the population based on their socio-economic situation. In addition, it seeks to ensure that state attention reaches as a

priority those people who are in extreme poverty, i.e., those who require urgent attention from the state to meet their most basic needs. The classification of the population in SISBEN is useful for decision-making and for targeting public policies and social programs of the attention of the local government (mayors' and governors' offices) and the national government.

The SISBEN must have accurate and updated information on the population that is most vulnerable due to its economic and financial situation. The inclusion in the SISBEN of people who do not require state assistance leads the state to make the wrong decisions when allocating resources, which directly affects the poorest.

The information consigned in Clear Accounts on Political Campaign Financiers in the elections of 2015 (territorial elections) and 2018 (legislative and presidential elections) and on the people who made contributions to finance the operation of political parties in the 2015-2019 period was reviewed. In this review, it was found that 34,075 campaign financiers in the period 2016 -2019 have contracts with the State, of this group.

There are 2,773 registered in SISBEN. In principle, this situation does not generate any problems because all citizens, regardless of their needs or economic capacity, have the right to finance political campaigns. However, it is impossible not to wonder about the financial capacity of these people if they are making contributions to political campaigns despite being SISBEN beneficiaries.

In this regard, among the contractor financiers registered in the SISBEN, 1,057 (approximately 38 percent) are classified with a score below 33 points. In other words, they are registered as extremely vulnerable. This condition qualifies them to access any state subsidy, whether for housing, education, conditional transfers, or health, among others. Even more serious is that of these 1057 people, 433 were in extreme poverty conditions in 2018 and 2019.

On the number of contributions to campaigns and political parties, the total of these resources amounts to 4,150 million 834,941 pesos. This means that on average each state contractor registered in the SISBEN contributed nine million 586,224 Colombian pesos. In addition to the generosity of these contributions, and despite their poverty status (according to the information reported in SISBEN), these individuals entered into contracts with the State for a total value of 118,693 million 713,265 pesos. That is, each person received on average a contract for a value of 274 million 119,430 pesos.

Emblematic cases

The following three cases show more clearly this relationship between funders - contractors and SISBEN beneficiaries.

- Davys Vallejo Petro registers contracts with the State for 101 million Colombian pesos and made contributions worth 40 million to Karen Quintero, Kareina Arteaga, and Rubén Darío Guerra, all candidates for the Córdoba assembly. Davis appears in the SISBEN database with a score of 3,37. This information was validated on June 5, 2019).
- José Gregorio Bayter Zumaqué appears registered in SISBEN with a score of 7.05, information validated on April 25, 2019. He has entered into contracts for 102 million 415,200 Colombian pesos and contributed 42 million 100,000 pesos to the campaign for the House of Representatives to Wadith Alberto Manzur and to the campaign for Mayor of Ayapel to Maricel Nader.
- Finally, Rubén Darío Gutiérrez Hoyos entered into contracts for 149 million 379,561 pesos and contributed 20 million pesos to the campaign of Marcos Pineda Garcia, elected mayor of Monteria Gutiérrez. is registered in the SISBEN base with a score of 7.37 points, information validated on November 26, 2018.

It is not understood how these people who according to the SISBEN deserve priority attention for being in a condition of poverty, manage to make significant contributions to various political campaigns and has also been contractors with the State. In the first place, this situation could be an indication that the attention of those who really need the SISBEN is being diverted to favor other people. Secondly, there is a disconnection between the different instances that collect relevant information on contracting and campaign financing, which do not share information. Thirdly, there is a more cultural and personal problem in which people who evidently have

favorable economic conditions seek to receive the benefits designed to support those who need it most, without caring that their criminal attitude is detrimental to those who are vulnerable and require urgent support from the State".

Proposals on the necessary reforms to the targeting tools of socioeconomic stratification and Sisbén in Colombia are beyond the scope of this paper, however, their full analysis is of vital importance for the fulfillment of the objectives of this work. Through the implementation of the project, the relevance of these instruments in the life, well-being, and mobilization opportunities of Colombians will be observed through the implementation of public policies that truly impact communities in need.

METHODOLOGY

The methodology chosen for the implementation of this project is based on the document "Fundamental Methodology for Data Science" (IBM, 2015), whose basis is CRISP-DM (Cross-Industry Standard Process for Data Mining) which can be considered as the de facto methodology for projects dedicated to extracting value from data, which has been a source of inspiration for other standards such as SEMMA of SAS or ASUM-DM (Institute of Knowledge Engineering, 2020). Implementing important advances such as the emphasis on several of the new practices in data science, such as the use of large volumes of data, the incorporation of text analytics in predictive modeling, and the automation of some processes (IBM, 2015).

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.

Aims of the project

General Objective

To analyze the relevance and correlation of targeting tools such as socio-economic stratification and the Sisbén 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.
- Generate a manual that allows students, teachers, IT professionals, and any citizen with notions about programming to validate, contrast and carry out their own analysis of the public policies generated in Colombia.
- Empower as many citizens as possible through knowledge, so that they can exercise their constitutional right to Citizen Oversight, which constitutes "the democratic mechanism of representation that allows citizens or different community organizations to exercise oversight over public management, with respect to the authorities, administrative, political, judicial, electoral, legislative and control bodies" (Law 80, 2003), with the aim of exercising proper oversight over projects aimed at overcoming inequality in Colombia, paying special attention to the choice of the target population.
- 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.

- Serve as a reference document for academic institutions and students in the IT area.

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, and 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

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

File_name: cityWorkCol.csv

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

Source 2:

https://www.datos.gov.co/en/Estad-sticas-Nacionales/Gran-Encuesta-Integrada-de-Hogares-GEIH/mcpt-3dws/data?no_mobile=true

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

2. Large Integrated Household Survey - GEIH (2021)

File_name: GEIH.csv

Source:

https://www.kaggle.com/datasets/catamelo/dane-geih-colombia-2021

General Description: The author of the dataset describes it as: This dataset consolidates the most important variables for the segment of Employed Wage-Earning Individuals (with a verbal or written contract with a company) belonging to the urban and main cities of Medellin, Barranquilla, Bogota, Bucaramanga, and Cali. This dataset consolidates the most important variables associated with the modules of:

1) General characteristics of persons

2) Employed persons

3) Households and Dwellings

4) Other activities and help during the week

5) Other income

Data Information

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

Language: Spanish

Geographical coverage: Departmental Capital City

Update Frequency: Unknown

Issue Date (yyyy-mm-dd): 2019-08-05

3. Poverty - Incidence Rate %

File_name: PovertyIncidenceRate.csv

Source:
https://www.bancomundial.org/es/topic/poverty/lac-equity-lab1/poverty/head-count

General Description: Almost a quarter of the LAC population lives on less than $5.5 USD-PPP 2011 a day, while slightly less than a third is considered middle class. This interactive chart presents trends in the poverty rate, the poverty gap, and the severity index under various poverty lines for LAC countries and sub-regions. The poverty lines include the World Bank's international extreme poverty line ($1.90-a-day in 2011 PPP) as well as the lines for lower-middle and upper-middle income countries, ($3.20 USD-2011 PPP) and poverty ($5.5 USD-PPP 2011). In addition, one can also choose to visualize the trends of the middle class and the vulnerable. Country data is further subdivided into rural and urban regions, where available.

Data Information

Owner of the dataset: World Bank

Language: Spanish

Geographical coverage: Latin American Region

Update Frequency: Unknown

Issue Date (yyyy): 2022

4. Poverty - Regional Distribution

File_name:

PovertyRegionalDistribution.csv

Source:
https://www.bancomundial.org/es/topic/poverty/lac-equity-lab1/poverty/regional-distribution

General Description: Brazil and Mexico are the most populous countries in the region and have almost half of the poor in the region. In contrast, less than 5 percent of the poor in LAC reside in the Southern Cone**.**

Data Information

Owner of the dataset: World Bank

Language: Spanish

Geographical coverage: Latin American Region

Update Frequency: Unknown

Issue Date (yyyy): 2020

5. Poverty - Mechanisms of change

File_name: PovertyMechanismsofchange.csv

Source:
https://www.bancomundial.org/es/topic/poverty/lac-equity-lab1/poverty/poverty-drivers

General Description: Over the past decade, changes in the LAC poverty rate have been driven primarily by economic growth, rather than changes in income distribution. This interactive graph shows the main drivers of changes in poverty in the region and by country, under different lines and for different indicators. A change in poverty can be driven by: (i) a change in average household income; and/or (ii) a change in the distribution of income. The factors can be negative or positive, depending on whether they decrease or increase poverty, respectively. The magnitude of the number will be useful in comparing which effect is more dominant than the other.

Data Information

Owner of the dataset: World Bank

Language: Spanish

Geographical coverage: Latin American Region

Update Frequency: Unknown

Issue Date (yyyy): 2017

6. Poverty - Income Contribution

File_name: PovertyIncomeContribution.csv

Source:
https://www.bancomundial.org/es/topic/poverty/lac-equity-lab1/poverty/contribution-of-income

General Description: Increased labor income has been the main driver of poverty reduction in LAC in the last decade. This interactive graph shows the contribution of income components to changes in poverty rates or the Gini. The results can be seen for different poverty lines, different periods, or by country, sub-region or for all of Latin America.

Data Information

Owner of the dataset: World Bank

Language: Spanish

Geographical coverage: Latin American Region

Update Frequency: Unknown

Issue Date (yyyy): 2022

7. Inequality - Income Distribution

File_name: DesigualdadDistribuciónDeIngresos.csv

Source:
https://www.bancomundial.org/es/topic/poverty/lac-equity-lab1/income-inequality/income-distribution

General Description: This dataset presents the income distribution at the national and regional levels in different years.

Data Information

Owner of the dataset: World Bank

Language: English

Geographical coverage: Latin American Region

Update Frequency: Unknown

Issue Date (yyyy): 2020

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.

The data comprehension process for each of the data sets is presented below.

From the definition of DataFrame: from the Kaggle platform's Pandas course: A DataFrame is a table. It contains an array of individual entries, each of which has a certain value. Each entry corresponds to a row (or record) and a column. The following analyses 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
```

Fig16: Loading modules and computer libraries

```
######### 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
```

Fig17. Loading modules and plotting libraries

1. File_name: cityWorkCol.csv

Dataframe creation

We implement the creation of the df1 corresponding to the cityWorkCol.csv file.

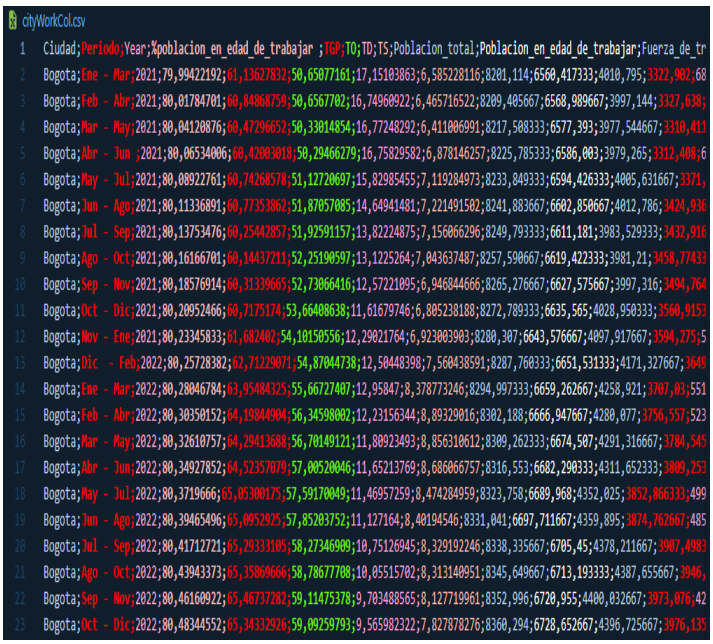


Fig18. cityWorkCol.csv.csv visualization in VSC

In data science, we usually analyze external data sets and many of these are in CSV format, so a CSV file is a table of comma-separated values. Hence the name: "Comma Separated Values" to read the data from a DataFrame we use the pd.read_csv() function.

The df.head(n) method is useful when you want to get a specific number of records, it returns the first n rows of the DataFrame. In this case the parameter given is 5, so an array with 5 records will be returned.

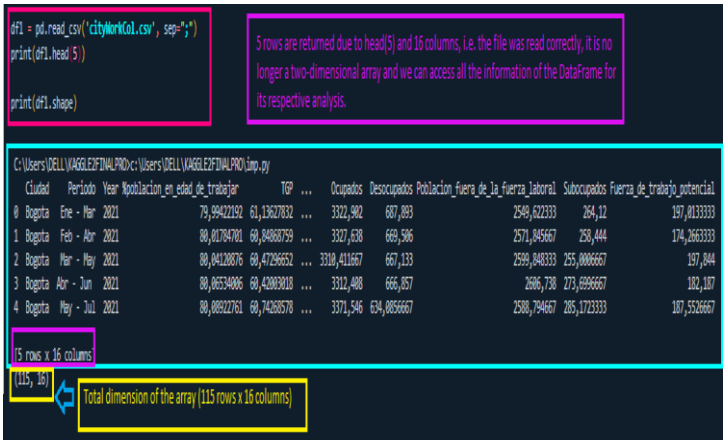


Fig19. Dataframe df1 creation with df.head() method

Sep is defined by the official Pandas documentation as a delimiter. In the process of reading the content of a file in csv format, the content of the dataframe is loaded and returned, when you look at the file for the first time you can notice that this is separated in this example by (;) and in fact if we run a df. shape the answer will be that this dataframe contains a single column that is to say it is arranged as a one-dimensional array and we cannot access its contents.

Therefore, so that the file can be read by pandas and can be manipulated it is necessary to apply this parameter to describe that in this case the delimiter of our data is (;). The separator can be another character or even a custom separator.

Below is the Dataframe generated without the application of sep.

Fig20. Dataframe sf1 visualization without sep parameter

In the previous image, it returns an array with the first 5 records generated by the parameter head(5). When observing the total dimensions of the array we have (115 rows x 1 column) it is a unidimensional array with 115 records. Under these conditions we cannot access the information contained in the file.

Next, you can see how the application of the parameter sep allows reading completely and correctly the file, in this way an array of sizes (115rows x 16columns). In this case the complete DataFrame is printed. You can see points arranged on the horizontal and vertical axes in the center of

the image, these points represent data that are not shown for visualization reasons, but that must be taken into account for the corresponding analysis.

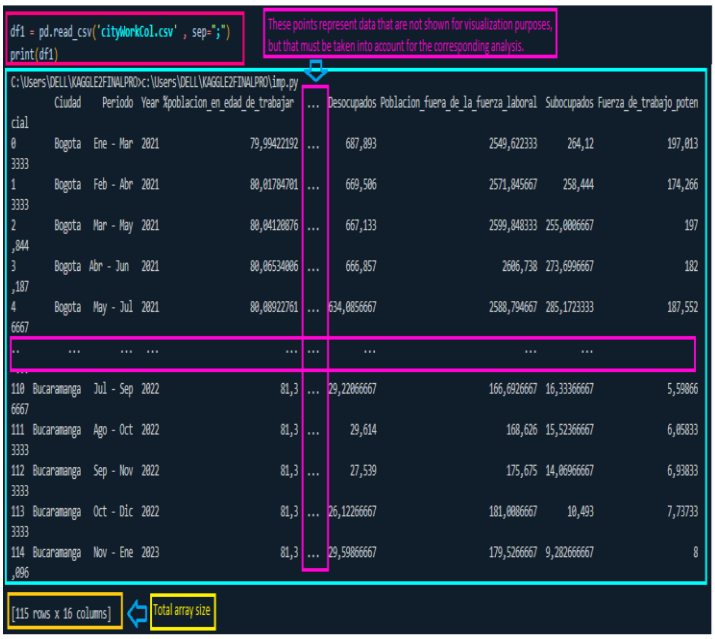


Fig20.Dataframe df1 visualization

It is also possible to access the last records of the array through the method df.tail(n) which returns the last n rows of the DataFrame in this case we access the last 6 records of the Dataframe.

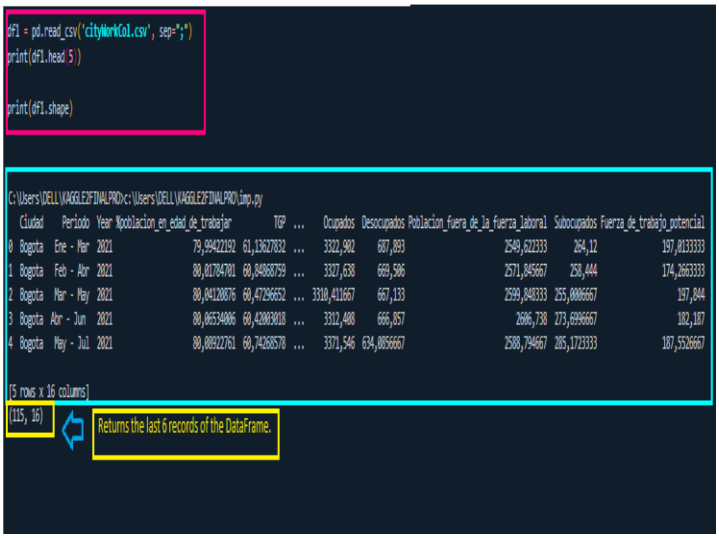


Fig21. Dataframe df1 visualization with df.tail() method

Total number of rows and columns

To obtain the total dimensions of the array, the df.shape method is applied, which returns a tuple with the number of rows and columns of the DataFrame.

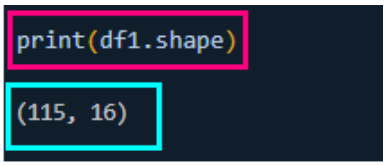


Fig22. Total number of rows and columns of dataframe df1

It is observed that the total dimension of the array is (115 rows x 16 columns). The first term corresponds to the number of rows or records and the second term corresponds to the number of columns of the DataFrame.

Specific selection of Dataframe values through set_option()

There are different ways to select specific values of a DataFrame, one of them being through the top-level options attribute.

We can select the number of rows and columns we want to print from the df. Although this is a limited indexing option it can be very useful when you have very large datasets and want to access a limited number of rows and columns.

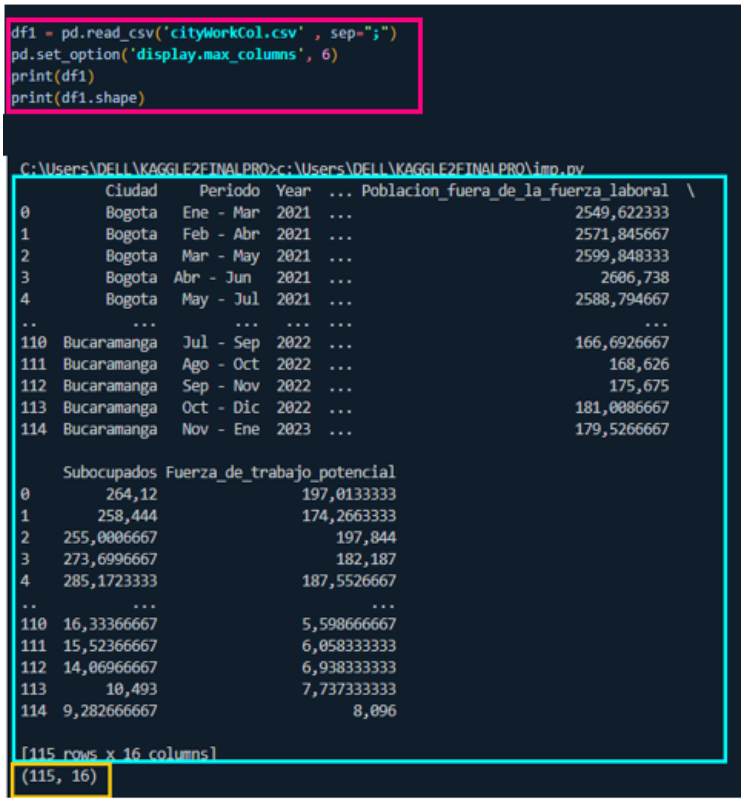


Fig23. Selection of specific columns values of df1 through set_option

It is important to understand that this is only a visualization process, the size of the array is not modified, when observing the output of the shape method we observe that the array continues with the dimensions of (115 rows x 16 columns).

Next we can choose both the number of records and columns that we want to obtain at the output

```
df1 = pd.read_csv('cityWorkCol.csv' , sep=";")
pd.set_option('display.max_columns', 6)
pd.set_option('display.max_rows', 5)
print(df1)
print(df1.shape)

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
          Ciudad    Periodo  Year  ... Poblacion_fuera_de_la_fuerza_laboral  \
0         Bogota  Ene - Mar  2021  ...                              2549,622333
1         Bogota  Feb - Abr  2021  ...                              2571,845667
..           ...        ...   ...  ...                                      ...
113  Bucaramanga  Oct - Dic  2022  ...                              181,0086667
114  Bucaramanga  Nov - Ene  2023  ...                              179,5266667

     Subocupados Fuerza_de_trabajo_potencial
0         264,12                197,0133333
1         258,444               174,2663333
..            ...                       ...
113        10,493               7,737333333
114  9,282666667                     8,096

[115 rows x 16 columns]
(115, 16)
```

Fig24. Selection of specific columns and rows values of df1 through set_option

General data information

We can obtain general information about our dataset through the df.info() attribute. It returns:

- The general information contained in each column includes its label.
- It informs us of the size of the Dataframe in this case it gives us a total of 16 columns and 115 non-null records.
- The number of non-null values in each column. In this example, we observe 115 non-null records.
- The type of data discriminated by column
- A small summary of grouping of data types in this case we observe that of a total of 16 columns only one is of type int64(1) and corresponds to 'Year', and the remaining 15 are of type object(15).
- Memory Used: in this case, it is 14.5+ KB

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

Data columns (total 16 columns):
 #   Column                           Non-Null Count  Dtype
---  ------                           --------------  -----
 0   Ciudad                           115 non-null    object
 1   Periodo                          115 non-null    object
 2   Year                             115 non-null    int64
 3   %poblacion_en_edad_de_trabajar   115 non-null    object
 4   TGP                              115 non-null    object
 5   TO                               115 non-null    object
 6   TD                               115 non-null    object
 7   TS                               115 non-null    object
 8   Poblacion_total                  115 non-null    object
 9   Poblacion_en_edad_de_trabajar    115 non-null    object
 10  Fuerza_de_trabajo                115 non-null    object
 11  Ocupados                         115 non-null    object
 12  Desocupados                      115 non-null    object
 13  Poblacion_fuera_de_la_fuerza_laboral  115 non-null  object
 14  Subocupados                      115 non-null    object
 15  Fuerza_de_trabajo_potencial      115 non-null    object
dtypes: int64(1), object(15)
memory usage: 14.5+ KB
None
```

Fig25. df.info() attribute of df1 DataFrame

To get a more detailed view of the column labels we apply the df.columns method.

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

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Index(['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', 'Subocupados',
       'Fuerza_de_trabajo_potencial'],
      dtype='object')
```

Fig26. df.columns attribute of df1 DataFrame

In this form, we obtain all the columns of the DataFrame.

Checking the data type

It is very useful to know what types of data we are using in a DataFrame, beyond establishing an overview of our data, it is possible that based on these results we should implement changes in the data types because there are certain operations within the DataFrame that are allowed between certain types of data.

```
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
```

Fig27. DataFrame df1 data types

We obtain 15 columns whose data type is object and an int64 data type corresponding to the Year column.

Request a list of unique values

The unique(). method returns a list of unique values corresponding to a specific column of the DataFrame. To apply this method, you must select the label corresponding to the column you want to analyze, in this case, it is the City label and applies the method. In this way, it performs a sweep of the entire column and extracts the non-repeated values found in it. The output returns the cities where the GEIH was implemented

```
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']
```

Fig28. unique() method

List of unique values and their frequency

When you want to see a list of unique values and the frequency with which they appear in the data set, use the value_counts() method, like the unique() method, select the label corresponding to the column you want to analyze, in this case the City label is selected again. The output shows the cities where the GEIH was implemented and in front of these the frequency with which they are repeated within the DataFrame.

```
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
```

Fig29. value_counts() method

List of unique values and their frequency

When you want to see a list of unique values and how often they appear in your dataset, use the value_counts() method, like the unique() method, and you select the label corresponding to the column you want to analyze, in this case, the City label is selected again. The cities where the GEIH was implemented are obtained at the output and in front of these the frequency with which they are repeated within the DataFrame.

The value_counts() method also allows selection to be implemented via the loc and iloc access operators.

The following figure shows the selection process through the loc operator, this paradigm bases the attribute selection process on labels. In this paradigm, it is the value of the data index, that is, its label, that is the selection criteria, not its position.

loc[:, ["TO", "TD"]] this range determines two values; first the value of the records or rows which is the value that is represented before the comma(,).

In this example, we have [:] which would be the range corresponding to the records, when we leave the spaces blank by default this range selects from row 0 to the last row.

After the comma (,) there is the range corresponding to the columns and here we have ["TO", "TD"], that is, we select two columns with the labels "TO" and "TD". These tags must always be enclosed in quotation marks and within a square parenthesis.

The corresponding output is presented below. You can see the generation of the list of unique values corresponding to the given selection parameter. In this case, a scan is made through all the records belonging to the columns identified with the labels "TO", and "TD", all the unique values found in this process are returned, and in front of each of them the value of its frequency. Data size and data type are also presented. in this case you have; length: 115 and dtype: int64

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
print(df1.loc[:, ["TO", "TD"]].value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
TO          TD
47,6        18,4          1
58,3        8,3           1
58,27346909 10,75126945   1
58,24277994 10,3514725    1
58,23822208 11,02642717   1
                         ..
52,63213843 11,2233033    1
52,6        16,4          1
52,5261467  14,09506988   1
52,52121832 13,63200887   1
61,93944781 9,30897634    1
Length: 115, dtype: int64
```

Fig30. value_counts() method with the loc operator

Next, the selection will be made through the selection operator iloc which selects data based on its numerical position.

In this case, the selected selection range is iloc[:,]. As in the previous case of the loc operator, the range determines two values; first, the value of the records or rows is the value that is represented before the comma (,) and then the value corresponding to the selected range of the columns. For this example, we are selecting all records from all columns. When we do not place any references inside the selection fields established by default, all the values are selected.

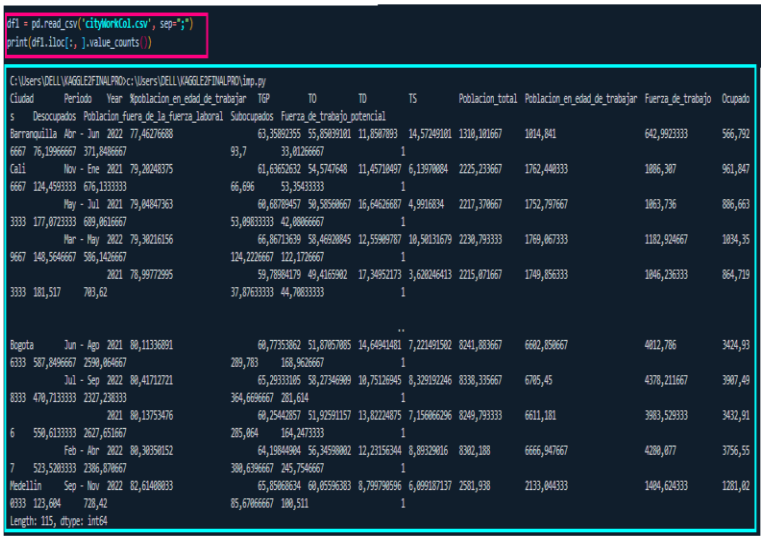


Fig31. value_counts() method with the iloc operator

It is relevant to analyze and compare the following outputs;

When observing the selection range of the first print, which corresponds to the previous implementation, it can be observed, as previously analyzed, that the given range establishes the selection of all the records and columns of the DataFrame, while the request within the second print does not present any selection operator. It requests that the value_counts() method be applied to the entire DataFrame, therefore, it can be inferred that the output corresponding to the two prints must be the same.

In conclusion, the two requests present the same selection range (the entire set of the DataFrame) written in two different ways.

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
print(df1.iloc[:, ].value_counts())
print(df1.value_counts())
```

Fig32. value_counts() method, selection of the entire dataset of the DataFrame

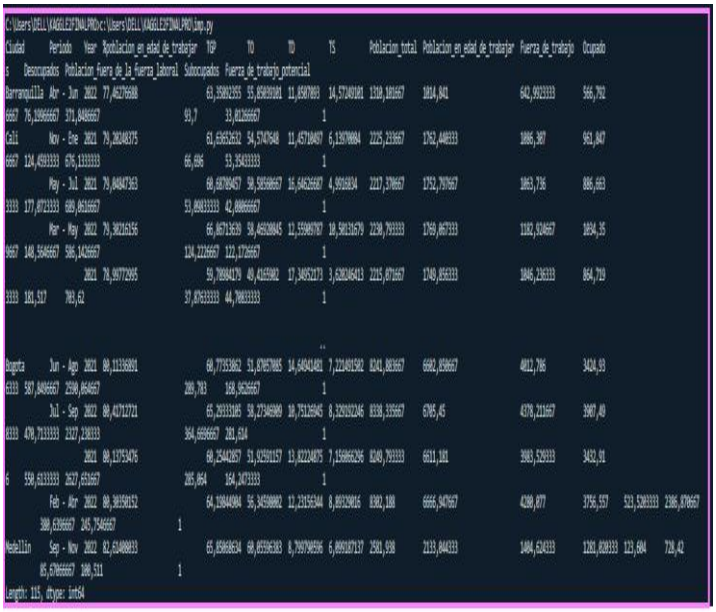


Fig33. value_counts() method, selection through (df1.iloc[:,].value_counts())

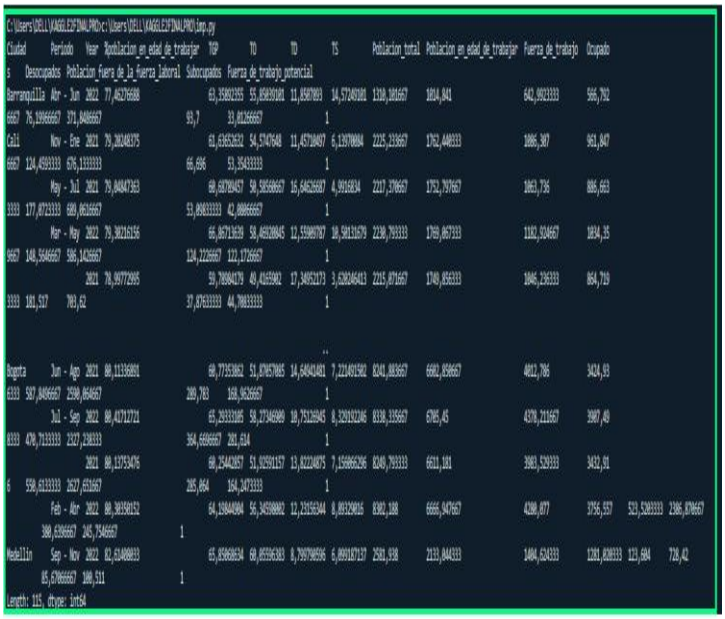


Fig34. value_counts() method, selection through (df1.value_counts())

Descriptive summary of the DataFrame df1

Descriptive summary of the DataFrame df1

A fundamental method in data analysis is df.describe(): which returns a DataFrame with a statistical summary of the array's columns. For numerical data (number) the mean, standard deviation, minimum, maximum, and quartiles are calculated. For non-numeric data (object) the number of values, the number of distinct values, the mode, and their frequency are calculated. It must be taken into account that if the type is not indicated, only numerical columns are considered.

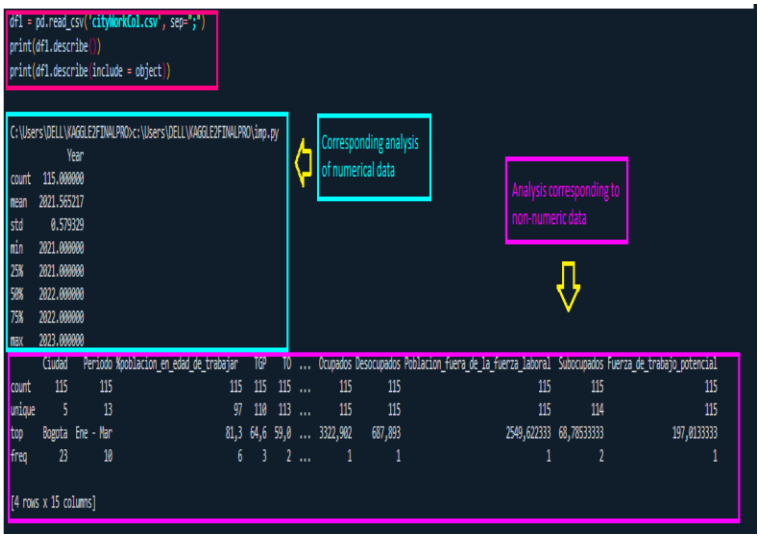


Fig35. Descriptive summary of the DataFrame df1

This method allows us to obtain valuable information about the DataFrame in which we are working, in this case, we can infer:

df.count(): Returns a series with the number of elements that are not null or NaN in each column of the DataFrame. In this case, 115 records are not null or NAN.

df.mean(): Returns a series with the means of the data of the numerical columns of the DataFrame, it is also known as average, it is the value obtained by dividing the sum of a conglomerate of numbers by the number of them. In the Year the mean column corresponds to 2021.565217.

df.std(): Returns a series with the standard deviations of the data in the numeric columns of the DataFrame. Std is a numerical index of the spread of a data set (or population). The greater the standard deviation, the greater the dispersion of the

population. The standard deviation is an average of the individual deviations of each observation from the mean of a distribution. In this data set, the std = 0.579329, this value indicates that the data in the set has a high rate of dispersion.

df.min(): Returns the smallest value of the data in the Year column, in this case, the value is 2021.

df.max(): Returns the largest value of the data in the Year column, in this case, the value is 2023.

The quartiles are statistical measures of position that have the property of dividing the statistical series into four groups of equal numbers of terms. Quartiles allow you to quickly assess the spread and central tendency of a data set.

Below is a table with the basic description of the quartiles.

Quartile	Description
1st quartile (Q1)	25% of the data is less than or equal to this value: 2021
2nd quartile (Q2)	The median. 50% of the data is less than or equal to this value: 2022
3rd quartile (Q3)	75% of the data is less than or equal to this value: 2022

Table6: Quartile description

In the case of object type data, we can see that the data corresponding to the previously analyzed count(), unique() methods, the frequency of occurrence, and the top value corresponding to the first data obtained from each column is generated.

Once the process corresponding to Stage 5: Understanding data with Dataframe 1 has been implemented and analyzed, the process is replicated in a general way for the other data sets.

2. File_name: **GEIH.csv**

Dataframe creation

We implement the creation of the df2 corresponding to the GEIH.csv file.

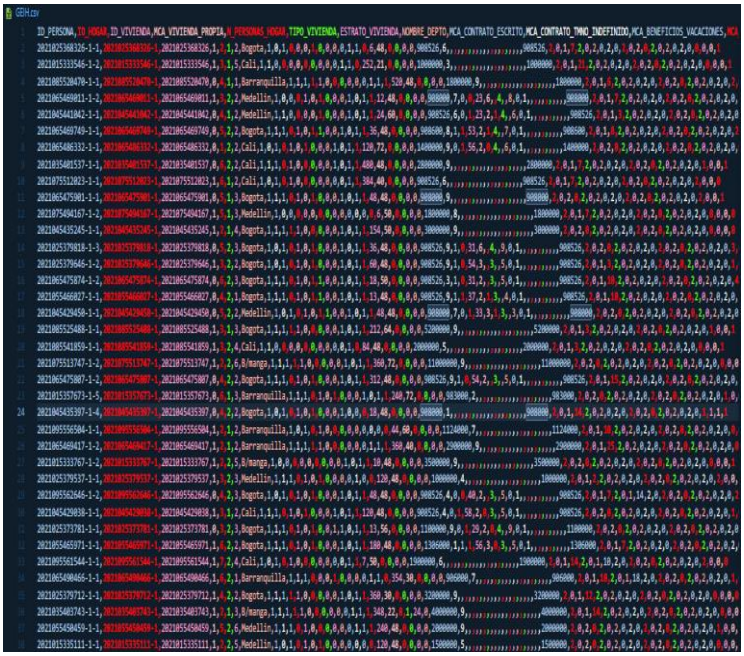


Fig36. GEIH.csv visualization in VSC

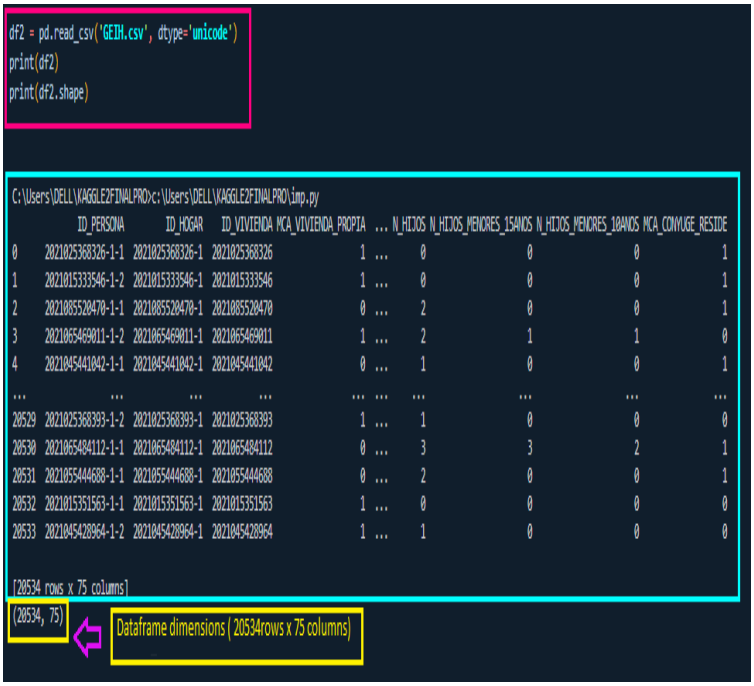


Fig37. Dataframe df2 visualization and df2 dimensions (20534 rows x 75 columns)

General data information

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode')
print(df2.info())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20534 entries, 0 to 20533
Data columns (total 75 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   ID_PERSONA                            20534 non-null  object
 1   ID_HOGAR                              20534 non-null  object
 2   ID_VIVIENDA                           20534 non-null  object
 3   MCA_VIVIENDA_PROPIA                   20534 non-null  object
 4   N_PERSONAS_HOGAR                      20534 non-null  object
 5   TIPO_VIVIENDA                         20534 non-null  object
 6   ESTRATO_VIVIENDA                      20534 non-null  object
 7   NOMBRE_DEPTO                          20534 non-null  object
 8   MCA_CONTRATO_ESCRITO                  20534 non-null  object
 9   MCA_CONTRATO_TMNO_INDEFINIDO          20534 non-null  object
 10  MCA_BENEFICIOS_VACACIONES             20534 non-null  object
 11  MCA_BENEFICIOS_PRIMA_NAVIDAD          20534 non-null  object
 12  MCA_BENEFICIOS_CESANTIAS              20534 non-null  object
 13  MCA_BENEFICIOS_AUX_ALIMENTACION       20534 non-null  object
 14  MCA_BENEFICIOS_SUB_TRANSPORTE         20534 non-null  object
 15  MCA_BENEFICIOS_SUB_FAMILIAR           20534 non-null  object
 16  MCA_BENEFICIOS_SUB_EDUCATIVO          20534 non-null  object
 17  MCA_BENEFICIOS_VIATICOS_BONIFICACIONES 20534 non-null  object
 18  MCA_FONDO_PENSIONES                   20534 non-null  object
 19  MCA_PENSIONADO                        20534 non-null  object
 20  MCA_AFILIADO_ARL                      20534 non-null  object
 21  MCA_CAJA_COMPENSACION                 20534 non-null  object
 22  ANTIGUEDAD_MESES                      20534 non-null  object
 23  HORAS_LABORALES_SEMANA                20534 non-null  object
 24  MCA_HR_EXTRA                          20534 non-null  object
 25  MCA_SEGUNDO_TRABAJO                   20534 non-null  object
 26  HORAS_SEGUNDO_TRABAJO_SEMANA          20534 non-null  object
 27  INGRESOS_HR_EXTRAS                    20534 non-null  object
 28  INGRESOS_LABORALES                    20534 non-null  object
 29  CATEGORIA_EMPLEADOS_EMPRESA           20534 non-null  object
 30  MCA_MUJER                             20249 non-null  object
 31  MCA_JEFE_HOGAR                        20249 non-null  object
 32  EDAD_ANOS                             20249 non-null  object
 33  ESTADO_CIVIL                          20249 non-null  object
 34  MCA_POBRE                             9085 non-null   object
 35  NIVEL_EDUCATIVO                       20249 non-null  object
 36  TITULO                                18351 non-null  object
 37  ANOS_ESCOLARIDAD                      20249 non-null  object
 38  MCA_ESTUDIA                           20249 non-null  object
 39  MCA_ANALFABETISMO                     20249 non-null  object
 40  OI_ARRIENDOS                          18391 non-null  object
 41  OI_CUOTA_ALIMENTARIA                  18391 non-null  object
 42  OI_AYUDA_GOBIERNO                     18391 non-null  object
 43  OI_AYUDA_EPRIVADAS                    18391 non-null  object
 44  OI_CESANTIAS                          18391 non-null  object
 45  OI_FAMILIA_ACCION                     18391 non-null  object
 46  OI_JOVENES_ACCION                     18391 non-null  object
 47  OI_COLOMBIA_MAYOR                     18391 non-null  object
 48  OI_OTRAS                              18391 non-null  object
 49  INGRESOS_ADICIONALES                  18391 non-null  object
 50  INGRESOS_FINALES                      20534 non-null  object
 51  CRIAR_ANIMALES                        20534 non-null  object
 52  HORAS_SEM_CRIA_ANIM                   20534 non-null  object
 53  OFICIOS_HOGAR                         20534 non-null  object
 54  HORAS_SEM_OFIC_HOGAR                  20534 non-null  object
 55  OFICIOS_OTROS                         20534 non-null  object
 56  HORAS_SEM_OFIC_OTROS                  20534 non-null  object
 57  CUIDAR_NINOS                          20534 non-null  object
 58  HORAS_SEM_CUI_NINOS                   20534 non-null  object
 59  CUIDAR_ANC_DISC                       20534 non-null  object
 60  HORAS_SEM_CUIDAR_ANC_DIS              20534 non-null  object
 61  ELAB_PRENDAS                          20534 non-null  object
 62  HORAS_SEM_ELAB_PRENDAS                20534 non-null  object
 63  ASISTIR_EVENTOS                       20534 non-null  object
 64  HORAS_SEM_ASIS_EVE                    20534 non-null  object
 65  AUTOC_VIVIENDA                        20534 non-null  object
 66  HORAS_SEM_AUT_VIVIENDA                20534 non-null  object
 67  TRAB_COMUNITARIO                      20534 non-null  object
 68  HORAS_SEM_TRAB_COM                    20534 non-null  object
 69  OTROS_TRAB_COMUNALES                  20534 non-null  object
 70  HORAS_SEM_OTROS_COMUNALES             20534 non-null  object
 71  N_HIJOS                               20534 non-null  object
 72  N_HIJOS_MENORES_15ANOS                20534 non-null  object
 73  N_HIJOS_MENORES_10ANOS                20534 non-null  object
 74  MCA_CONYUGE_RESIDE                    20534 non-null  object
dtypes: object(75)
memory usage: 11.7+ MB
None
```

Fig38. df.info() attribute of df2 DataFrame

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode')
print(df2.columns)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Index(['ID_PERSONA', 'ID_HOGAR', 'ID_VIVIENDA', 'MCA_VIVIENDA_PROPIA',
       'N_PERSONAS_HOGAR', 'TIPO_VIVIENDA', 'ESTRATO_VIVIENDA', 'NOMBRE_DEPTO',
       'MCA_CONTRATO_ESCRITO', 'MCA_CONTRATO_TMNO_INDEFINIDO',
       'MCA_BENEFICIOS_VACACIONES', 'MCA_BENEFICIOS_PRIMA_NAVIDAD',
       'MCA_BENEFICIOS_CESANTIAS', 'MCA_BENEFICIOS_AUX_ALIMENTACION',
       'MCA_BENEFICIOS_SUB_TRANSPORTE', 'MCA_BENEFICIOS_SUB_FAMILIAR',
       'MCA_BENEFICIOS_SUB_EDUCATIVO',
       'MCA_BENEFICIOS_VIATICOS_BONIFICACIONES', 'MCA_FONDO_PENSIONES',
       'MCA_PENSIONADO', 'MCA_AFILIADO_ARL', 'MCA_CAJA_COMPENSACION',
       'ANTIGUEDAD_MESES', 'HORAS_LABORALES_SEMANA', 'MCA_HR_EXTRA',
       'MCA_SEGUNDO_TRABAJO', 'HORAS_SEGUNDO_TRABAJO_SEMANA',
       'INGRESOS_HR_EXTRAS', 'INGRESOS_LABORALES',
       'CATEGORIA_EMPLEADOS_EMPRESA', 'MCA_MUJER', 'MCA_JEFE_HOGAR',
       'EDAD_ANOS', 'ESTADO_CIVIL', 'MCA_POBRE', 'NIVEL_EDUCATIVO', 'TITULO',
       'ANOS_ESCOLARIDAD', 'MCA_ESTUDIA', 'MCA_ANALFABETISMO', 'OI_ARRIENDOS',
       'OI_CUOTA_ALIMENTARIA', 'OI_AYUDA_GOBIERNO', 'OI_AYUDA_EPRIVADAS',
       'OI_CESANTIAS', 'OI_FAMILIA_ACCION', 'OI_JOVENES_ACCION',
       'OI_COLOMBIA_MAYOR', 'OI_OTRAS', 'INGRESOS_ADICIONALES',
       'INGRESOS_FINALES', 'CRIAR_ANIMALES', 'HORAS_SEM_CRIA_ANIM',
       'OFICIOS_HOGAR', 'HORAS_SEM_OFIC_HOGAR', 'OFICIOS_OTROS',
       'HORAS_SEM_OFIC_OTROS', 'CUIDAR_NINOS', 'HORAS_SEM_CUI_NINOS',
       'CUIDAR_ANC_DISC', 'HORAS_SEM_CUIDAR_ANC_DIS', 'ELAB_PRENDAS',
       'HORAS_SEM_ELAB_PRENDAS', 'ASISTIR_EVENTOS', 'HORAS_SEM_ASIS_EVE',
       'AUTOC_VIVIENDA', 'HORAS_SEM_AUT_VIVIENDA', 'TRAB_COMUNITARIO',
       'HORAS_SEM_TRAB_COM', 'OTROS_TRAB_COMUNALES',
       'HORAS_SEM_OTROS_COMUNALES', 'N_HIJOS', 'N_HIJOS_MENORES_15ANOS',
       'N_HIJOS_MENORES_10ANOS', 'MCA_CONYUGE_RESIDE'],
      dtype='object')
```

Fig39. df.columns attribute of df2 DataFrame

Checking the data type

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode')
print(df2.dtypes)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
ID_PERSONA                object
ID_HOGAR                  object
ID_VIVIENDA               object
MCA_VIVIENDA_PROPIA       object
N_PERSONAS_HOGAR          object
                           ...
HORAS_SEM_OTROS_COMUNALES object
N_HIJOS                   object
N_HIJOS_MENORES_15ANOS    object
N_HIJOS_MENORES_10ANOS    object
MCA_CONYUGE_RESIDE        object
Length: 75, dtype: object
```

Fig40. DataFrame df2 data types

Request a list of unique values

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode')
print(df2.N_PERSONAS_HOGAR.unique())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
['2' '3' '4' '5' '1' '6' '7' '9' '8' '10' '12' '23' '16' '11' '13' '17'
 '14' '18' '19' '15']
```

Fig41. unique() method

List of unique values and their frequency

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode')
print(df2.N_PERSONAS_HOGAR.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
3     5374
4     5080
2     3684
5     2523
1     1369
6     1204
7      580
8      309
9      180
10     103
11      39
12      39
13      14
14      11
16       8
23       6
15       5
17       2
18       2
19       2
Name: N_PERSONAS_HOGAR, dtype: int64
```

Fig42. value_counts() method for N_PERSONAS_HOGAR column

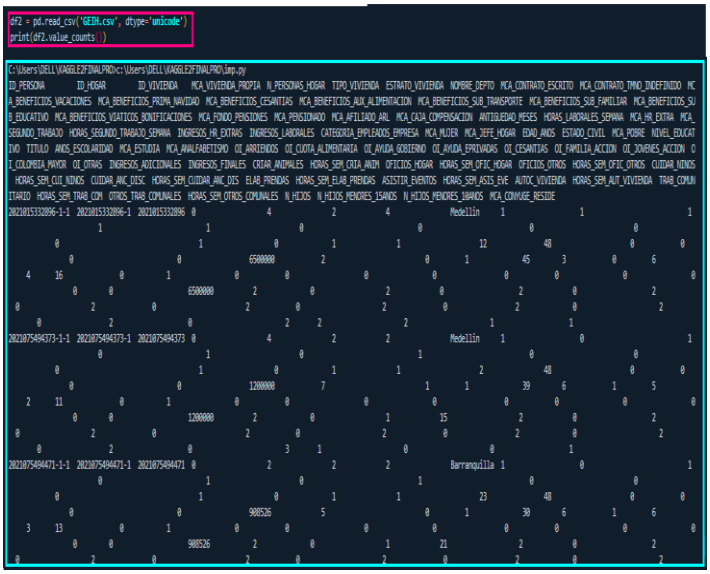


Fig43a. value_counts() method for df2 DataFrame Part1

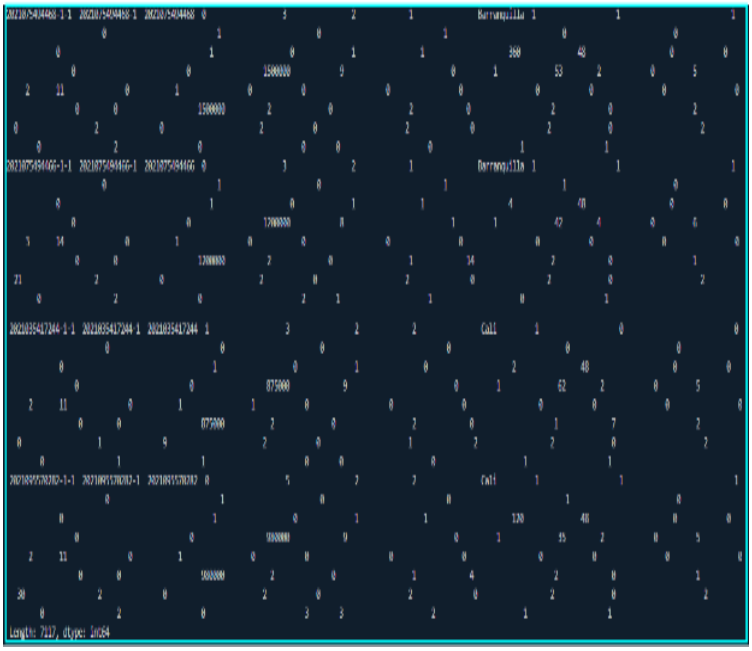


Fig43b. value_counts() method for df2 DataFrame Part2

Descriptive summary of the DataFrame df2

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode')
print(df2.describe())
print(df2.describe(include = object))
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
       ID_PERSONA       ID_HOGAR  ID_VIVIENDA MCA_VIVIENDA_PROPIA  ... N_HIJOS N_HIJOS_MENORES_15ANOS N_HIJOS_MENORES_10ANOS MCA_CONYUGE_RESIDE
count       20534          20534        20534                20534  ...   20534                 20534                 20534               20534
unique      20534          15492        15392                    2  ...      10                     7                     5                   2
top    2021025368326-1-1 2021095556450-1 2021095551531            0  ...       1                     0                     0                   1
freq            1              6            6                12331  ...    7036                 13558                 15642               12962

[4 rows x 75 columns]
       ID_PERSONA       ID_HOGAR  ID_VIVIENDA MCA_VIVIENDA_PROPIA  ... N_HIJOS N_HIJOS_MENORES_15ANOS N_HIJOS_MENORES_10ANOS MCA_CONYUGE_RESIDE
count       20534          20534        20534                20534  ...   20534                 20534                 20534               20534
unique      20534          15492        15392                    2  ...      10                     7                     5                   2
top    2021025368326-1-1 2021095556450-1 2021095551531            0  ...       1                     0                     0                   1
freq            1              6            6                12331  ...    7036                 13558                 15642               12962

[4 rows x 75 columns]
```

Fig44. Descriptive summary of the DataFrame df2

3. File_name: PovertyIncidenceRate.csv

Dataframe creation

We implement the creation of the df3 corresponding to the PovertyIncidenceRate.csv file.

PovertyIncidenceRate.csv

```
Area;País;Línea Pobreza;Indicador;Año;Tasa
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2003;13,2
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2004;12,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2002;12,2
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2000;12
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2001;11,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2005;11,6
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2006;10,4
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2020;10,2
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2007;8,7
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2008;8,6
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2009;8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2012;7,9
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2010;7,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2011;7,6
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2018;7
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2013;6,9
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2019;6,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2016;6,7
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2017;6,6
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2014;6,6
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2015;6,3
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2003;5,3
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2004;4,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2002;4,7
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2001;4,7
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2000;4,7
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2005;4,5
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2020;4,1
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2006;3,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Severidad de la pobreza;2003;3,1
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2007;2,9
Nacional;América Central;Pobreza $2.15 (2017 PPP);Brecha de pobreza;2008;2,8
```

Fig45. PovertyIncidenceRate.csv visualization in VSC

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3)
print(df3.shape)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
          Area           País                 Línea Pobreza       Indicador   Año Tasa
0     Nacional  América Central       Pobreza $2.15 (2017 PPP)  Tasa de pobreza  2003  13,2
1     Nacional  América Central       Pobreza $2.15 (2017 PPP)  Tasa de pobreza  2004  12,8
2     Nacional  América Central       Pobreza $2.15 (2017 PPP)  Tasa de pobreza  2002  12,2
3     Nacional  América Central       Pobreza $2.15 (2017 PPP)  Tasa de pobreza  2000    12
4     Nacional  América Central       Pobreza $2.15 (2017 PPP)  Tasa de pobreza  2001  11,8
...        ...             ...                            ...              ...   ...   ...
10621    Rural         Uruguay  Clase media $14-$81 (2017 PPP)  Tasa de pobreza  2004   NaN
10622    Rural         Uruguay  Clase media $14-$81 (2017 PPP)  Tasa de pobreza  2003   NaN
10623    Rural         Uruguay  Clase media $14-$81 (2017 PPP)  Tasa de pobreza  2002   NaN
10624    Rural         Uruguay  Clase media $14-$81 (2017 PPP)  Tasa de pobreza  2001   NaN
10625    Rural         Uruguay  Clase media $14-$81 (2017 PPP)  Tasa de pobreza  2000   NaN

[10626 rows x 6 columns]
(10626, 6)
```

Dataframe dimensions (10626 rows x 6 columns)

Fig46. Dataframe df3 visualization and df3 dimensions (10626 rows x 6 columns)

General data information

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3.info())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10626 entries, 0 to 10625
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Area           10626 non-null  object
 1   País           10626 non-null  object
 2   Línea Pobreza  10626 non-null  object
 3   Indicador      10626 non-null  object
 4   Año            10626 non-null  int64
 5   Tasa           10373 non-null  object
dtypes: int64(1), object(5)
memory usage: 498.2+ KB
None
```

Fig47. df.info() attribute of df3 DataFrame

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3.columns)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Index(['Area', 'País', 'Línea Pobreza', 'Indicador', 'Año', 'Tasa'], dtype='object')
```

Fig48. df.columns attribute of df3 DataFrame

Checking the data type

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3.dtypes)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Area            object
País            object
Línea Pobreza   object
Indicador       object
Año              int64
Tasa            object
dtype: object
```

Fig49. DataFrame df3 data types

Request a list of unique values

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3.País.unique())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
['América Central' 'América Latina y el Caribe' 'Argentina (urbano)'
 'Bolivia' 'Brasil' 'Brasil-PNADC' 'Chile' 'Colombia' 'Cono Sur'
 'Costa Rica' 'Ecuador' 'El Salvador' 'Guatemala' 'Honduras' 'México'
 'Nicaragua' 'Panamá' 'Paraguay' 'Perú' 'Región Andina'
 'República Dominicana' 'St. Lucia' 'Uruguay']
```

Fig50. unique() method

List of unique values and their frequency

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3.País.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Uruguay                  693
Costa Rica               693
República Dominicana     693
Perú                     693
América Latina y el Caribe  693
Paraguay                 660
Panamá                   660
El Salvador              660
Honduras                 627
Colombia                 594
Bolivia                  594
Argentina (urbano)       561
Ecuador                  462
Brasil                   462
México                   396
Brasil-PNADC             297
Cono Sur                 231
Chile                    231
Región Andina            231
América Central          231
Nicaragua                132
Guatemala                 99
St. Lucia                 33
Name: País, dtype: int64
```

Fig51. value_counts() method for País column

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Area      País             Línea Pobreza                 Indicador                Año   Tasa
Nacional  América Central  Clase media $14-$81 (2017 PPP)  Tasa de pobreza          2000  21,9   1
Rural     Uruguay          Pobreza $2.15 (2017 PPP)        Severidad de la pobreza  2008  0,1    1
                           Clase media $14-$81 (2017 PPP)  Tasa de pobreza          2020  62,7   1
                           Pobreza $2.15 (2017 PPP)        Brecha de pobreza        2006  0,2    1
                                                                                    2007  0,1    1
                                                                                                  ..
Nacional  Región Andina    Pobreza $6.85 (2017 PPP)        Brecha de pobreza        2012  14,8   1
                                                                                    2013  13,8   1
                                                                                    2014  12,8   1
                                                                                    2015  12,4   1
Urbano    Uruguay          Vulnerable $6.85-$14 (2017 PPP) Tasa de pobreza          2020  23,7   1
Length: 10373, dtype: int64
```

Fig52. value_counts() method for df3 DataFrame

Descriptive summary of the DataFrame df3

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3.describe())
print(df3.describe(include = object))
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
                Año
count  10626.000000
mean    2010.360248
std        5.863153
min     2000.000000
25%     2005.000000
50%     2011.000000
75%     2015.000000
max     2020.000000
            Area     País             Línea Pobreza       Indicador   Tasa
count      10626    10626                     10626           10626  10373
unique         3       23                         5               3    717
top     Nacional  Uruguay  Pobreza $2.15 (2017 PPP)  Tasa de pobreza    0,2
freq        4004      693                      2898            4830    185
```

Fig53. Descriptive summary of the DataFrame df3

4.File_name: PovertyIncomeContribution.csv

We implement the creation of the df4 corresponding to the PovertyRegionalDistribution.csv file.

Dataframe creation

PovertyRegionalDistribution.csv

```
Tipo;País;Year;Línea de Pobreza;Share
Por país;Argentina (urbano);2002;Pobreza $6.85 (2017 PPP);3,3%
Por país;Argentina (urbano);2003;Pobreza $6.85 (2017 PPP);3,1%
Por país;Argentina (urbano);2002;Pobreza $3.65 (2017 PPP);2,9%
Por país;Argentina (urbano);2001;Pobreza $6.85 (2017 PPP);2,8%
Por país;Argentina (urbano);2002;Pobreza $2.15 (2017 PPP);2,7%
Por país;Argentina (urbano);2020;Pobreza $6.85 (2017 PPP);2,6%
Por país;Argentina (urbano);2004;Pobreza $6.85 (2017 PPP);2,6%
Por país;Argentina (urbano);2000;Pobreza $6.85 (2017 PPP);2,6%
Por país;Argentina (urbano);2003;Pobreza $3.65 (2017 PPP);2,6%
Por país;Argentina (urbano);2003;Pobreza $2.15 (2017 PPP);2,4%
Por país;Argentina (urbano);2005;Pobreza $6.85 (2017 PPP);2,3%
Por país;Argentina (urbano);2019;Pobreza $6.85 (2017 PPP);2,1%
Por país;Argentina (urbano);2004;Pobreza $3.65 (2017 PPP);2,1%
Por país;Argentina (urbano);2001;Pobreza $3.65 (2017 PPP);2,1%
Por país;Argentina (urbano);2007;Pobreza $6.85 (2017 PPP);2,0%
Por país;Argentina (urbano);2006;Pobreza $6.85 (2017 PPP);2,0%
Por país;Argentina (urbano);2000;Pobreza $3.65 (2017 PPP);1,9%
Por país;Argentina (urbano);2004;Pobreza $2.15 (2017 PPP);1,9%
Por país;Argentina (urbano);2001;Pobreza $2.15 (2017 PPP);1,9%
Por país;Argentina (urbano);2018;Pobreza $6.85 (2017 PPP);1,8%
Por país;Argentina (urbano);2008;Pobreza $6.85 (2017 PPP);1,8%
Por país;Argentina (urbano);2020;Pobreza $3.65 (2017 PPP);1,8%
Por país;Argentina (urbano);2005;Pobreza $3.65 (2017 PPP);1,8%
Por país;Argentina (urbano);2009;Pobreza $6.85 (2017 PPP);1,7%
Por país;Argentina (urbano);2000;Pobreza $2.15 (2017 PPP);1,7%
Por país;Argentina (urbano);2010;Pobreza $6.85 (2017 PPP);1,6%
Por país;Argentina (urbano);2006;Pobreza $3.65 (2017 PPP);1,6%
```

Fig54. PovertyRegionalDistribution.csv visualization in VSC

```
df4 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
print(df4)
print(df4.shape)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
            Tipo              País  Year         Línea de Pobreza  Share
0       Por país  Argentina (urbano)  2002  Pobreza $6.85 (2017 PPP)   3,3%
1       Por país  Argentina (urbano)  2003  Pobreza $6.85 (2017 PPP)   3,1%
2       Por país  Argentina (urbano)  2002  Pobreza $3.65 (2017 PPP)   2,9%
3       Por país  Argentina (urbano)  2001  Pobreza $6.85 (2017 PPP)   2,8%
4       Por país  Argentina (urbano)  2002  Pobreza $2.15 (2017 PPP)   2,7%
...          ...                ...   ...                       ...    ...
1381  Sub-regional      Región Andina  2014  Pobreza $6.85 (2017 PPP)  21,3%
1382  Sub-regional      Región Andina  2018  Pobreza $3.65 (2017 PPP)  21,1%
1383  Sub-regional      Región Andina  2017  Pobreza $2.15 (2017 PPP)  20,2%
1384  Sub-regional      Región Andina  2018  Pobreza $2.15 (2017 PPP)  18,7%
1385  Sub-regional      Región Andina  2019  Pobreza $2.15 (2017 PPP)  18,6%

[1386 rows x 5 columns]
(1386, 5)
```

Dataframe dimensions (1386 rows x 5 columns)

Fig55. Dataframe df4 visualization and df4 dimensions (1386 rows x 5 columns)

General data information

```
df4 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
print(df4.info())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1386 entries, 0 to 1385
Data columns (total 5 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Tipo              1386 non-null   object
 1   País              1386 non-null   object
 2   Year              1386 non-null   int64
 3   Línea de Pobreza  1386 non-null   object
 4   Share             1386 non-null   object
dtypes: int64(1), object(4)
memory usage: 54.3+ KB
None
```

Fig56. df.info() attribute of df4 DataFrame

```
df4 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
print(df4.columns)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Index(['Tipo', 'País', 'Year', 'Línea de Pobreza', 'Share'], dtype='object')
```

Fig57. df.columns attribute of df4 DataFrame

Checking the data type

```
df4 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
print(df4.dtypes)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Tipo              object
País              object
Year               int64
Línea de Pobreza  object
Share             object
dtype: object
```

Fig58. DataFrame df4 data types

Request a list of unique values

```
df4 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
print(df4.Share.unique())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
['3,3%' '3,1%' '2,9%' '2,8%' '2,7%' '2,6%' '2,4%' '2,3%' '2,1%' '2,0%'
 '1,9%' '1,8%' '1,7%' '1,6%' '1,5%' '1,4%' '1,3%' '1,2%' '1,1%' '1,0%'
 '0,9%' '0,8%' '0,7%' '0,5%' '3,6%' '3,2%' '2,5%' '2,2%' '48,2%' '47,6%'
 '47,5%' '44,4%' '42,8%' '42,7%' '42,2%' '41,7%' '41,4%' '41,2%' '41,1%'
 '41,0%' '40,8%' '40,5%' '40,3%' '40,1%' '40,0%' '39,9%' '39,8%' '39,6%'
 '39,4%' '39,0%' '38,9%' '38,7%' '38,5%' '38,4%' '38,3%' '38,1%' '38,0%'
 '37,8%' '37,6%' '37,4%' '37,0%' '36,9%' '36,6%' '36,4%' '36,0%' '35,9%'
 '35,7%' '35,3%' '35,2%' '34,7%' '34,4%' '34,3%' '34,0%' '33,5%' '33,4%'
 '33,3%' '32,1%' '32,0%' '31,0%' '30,9%' '25,4%' '20,0%' '18,2%' '3,0%'
 '0,6%' '0,4%' '0,3%' '20,3%' '16,9%' '14,1%' '13,4%' '13,2%' '13,1%'
 '13,0%' '12,7%' '12,6%' '12,5%' '12,3%' '12,1%' '12,0%' '11,9%' '11,8%'
 '11,7%' '11,6%' '11,4%' '11,3%' '11,2%' '11,1%' '11,0%' '10,9%' '10,8%'
 '10,7%' '10,6%' '10,5%' '10,4%' '10,3%' '10,1%' '10,0%' '9,9%' '9,8%'
 '9,5%' '9,1%' '0,2%' '5,1%' '4,5%' '4,2%' '4,0%' '3,9%' '3,8%' '3,7%'
 '3,5%' '3,4%' '13,5%' '8,9%' '8,8%' '8,4%' '8,2%' '8,0%' '7,6%' '7,5%'
 '7,4%' '7,2%' '7,1%' '7,0%' '6,6%' '6,5%' '6,4%' '6,1%' '6,0%' '5,9%'
 '5,7%' '5,6%' '5,5%' '5,4%' '5,0%' '4,8%' '4,4%' '4,3%' '4,1%' '8,5%'
 '5,8%' '5,3%' '5,2%' '4,6%' '28,9%' '27,1%' '26,7%' '26,3%' '25,6%'
 '25,2%' '25,0%' '24,8%' '24,7%' '24,4%' '24,3%' '24,1%' '23,9%' '22,5%'
 '22,4%' '22,3%' '22,2%' '22,1%' '21,6%' '21,5%' '21,1%' '21,0%' '20,8%'
 '20,5%' '19,9%' '19,5%' '19,3%' '19,1%' '18,4%' '18,3%' '18,0%' '17,8%'
 '17,5%' '17,4%' '17,2%' '17,1%' '16,5%' '16,4%' '15,9%' '15,8%' '15,4%'
 '15,3%' '15,2%' '14,4%' '14,2%' '10,2%' '9,7%' '9,6%' '9,4%' '9,0%'
 '8,6%' '8,3%' '8,1%' '7,9%' '7,8%' '7,7%' '7,3%' '6,9%' '6,8%' '6,7%'
 '6,2%' '0,1%' '0,0%' '26,9%' '21,9%' '17,9%' '17,6%' '17,3%' '16,8%'
 '16,6%' '16,3%' '16,2%' '15,7%' '15,5%' '15,1%' '14,9%' '14,8%' '14,5%'
 '14,0%' '13,9%' '13,6%' '13,3%' '12,8%' '4,9%' '4,7%' '35,5%' '33,0%'
 '29,9%' '29,6%' '29,4%' '29,3%' '29,1%' '28,4%' '28,1%' '27,8%' '27,7%'
 '27,5%' '26,8%' '26,2%' '26,1%' '25,8%' '25,7%' '25,3%' '24,9%' '24,6%'
 '23,8%' '23,6%' '23,5%' '23,3%' '23,2%' '23,1%' '23,0%' '22,9%' '22,7%'
 '22,0%' '21,8%' '21,7%' '21,3%' '20,2%' '18,7%' '18,6%']
```

Fig59. unique() method

List of unique values and their frequency

```
df4 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
print(df4.Share.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
0,4%     54
0,3%     50
1,6%     44
1,4%     35
0,5%     34
        ..
16,6%     1
16,8%     1
17,6%     1
17,9%     1
18,6%     1
Name: Share, Length: 285, dtype: int64
```

Fig60. value_counts() method for Share column

```
df4 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
print(df4.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Tipo          País               Year  Línea de Pobreza          Share
Por país      Argentina (urbano) 2000  Pobreza $2.15 (2017 PPP)  1,7%     1
              Perú               2012  Pobreza $6.85 (2017 PPP)  6,6%     1
                                 2015  Pobreza $3.65 (2017 PPP)  7,4%     1
                                       Pobreza $2.15 (2017 PPP)  7,4%     1
                                 2014  Pobreza $6.85 (2017 PPP)  6,7%     1
                                                                         ..
              El Salvador        2006  Pobreza $6.85 (2017 PPP)  1,8%     1
                                       Pobreza $3.65 (2017 PPP)  1,6%     1
                                       Pobreza $2.15 (2017 PPP)  1,4%     1
                                 2005  Pobreza $6.85 (2017 PPP)  1,7%     1
Sub-regional  Región Andina      2020  Pobreza $6.85 (2017 PPP)  27,7%    1
Length: 1386, dtype: int64
```

Fig61. value_counts() method for df4 DataFrame

Descriptive summary of the DataFrame df4

```
df4 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
print(df4.describe())
print(df4.describe(include = object))
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
              Year
count  1386.000000
mean   2010.000000
std       6.057486
min    2000.000000
25%    2005.000000
50%    2010.000000
75%    2015.000000
max    2020.000000
           Tipo    País          Línea de Pobreza Share
count      1386    1386                      1386  1386
unique        2      20                         3   285
top     Por país  México  Pobreza $6.85 (2017 PPP)  0,4%
freq       1071     126                       462    54
```

Fig62. Descriptive summary of the DataFrame df4

5.File_name:

PovertyMechanismsofchange.csv

Dataframe creation

We implement the creation of the df5 corresponding to the PovertyIncidenceRate.csv file.

PovertyMechanismsofchange.csv

```
País;Indicador;Línea de pobreza;Años Gap;Componente;Tasa
América Central;Brecha de pobreza;Clase media $16-$81 (2017 PPP);2010-2015;Redistribución;0
América Central;Brecha de pobreza;Pobreza $2.15 (2017 PPP);2010-2015;Redistribución;-0,1
América Central;Brecha de pobreza;Pobreza $3.65 (2017 PPP);2010-2015;Redistribución;-0,3
América Central;Brecha de pobreza;Pobreza $2.15 (2017 PPP);2010-2015;Crecimiento;-0,4
América Central;Brecha de pobreza;Pobreza $2.15 (2017 PPP);2010-2015;Total;-0,5
América Central;Brecha de pobreza;Vulnerable $6.85-$14 (2017 PPP);2010-2015;Redistribución;-0,5
América Central;Brecha de pobreza;Pobreza $6.85 (2017 PPP);2010-2015;Redistribución;-0,7
América Central;Brecha de pobreza;Pobreza $3.65 (2017 PPP);2010-2015;Crecimiento;-1,1
América Central;Brecha de pobreza;Clase media $16-$81 (2017 PPP);2010-2015;Total;-1,2
América Central;Brecha de pobreza;Clase media $16-$81 (2017 PPP);2010-2015;Crecimiento;-1,3
América Central;Brecha de pobreza;Pobreza $3.65 (2017 PPP);2010-2015;Total;-1,4
América Central;Brecha de pobreza;Pobreza $6.85 (2017 PPP);2010-2015;Crecimiento;-2,2
América Central;Brecha de pobreza;Pobreza $6.85 (2017 PPP);2010-2015;Total;-2,9
América Central;Brecha de pobreza;Vulnerable $6.85-$14 (2017 PPP);2010-2015;Crecimiento;-3
América Central;Brecha de pobreza;Vulnerable $6.85-$14 (2017 PPP);2010-2015;Total;-3,5
América Central;Brecha de pobreza;Pobreza $3.65 (2017 PPP);2010-2020;Redistribución;2,5
América Central;Brecha de pobreza;Pobreza $2.15 (2017 PPP);2010-2020;Redistribución;2,1
América Central;Brecha de pobreza;Pobreza $2.15 (2017 PPP);2010-2020;Total;1,7
América Central;Brecha de pobreza;Pobreza $3.65 (2017 PPP);2010-2020;Total;1,6
América Central;Brecha de pobreza;Pobreza $6.85 (2017 PPP);2010-2020;Redistribución;1,5
América Central;Brecha de pobreza;Vulnerable $6.85-$14 (2017 PPP);2010-2020;Redistribución;0,3
América Central;Brecha de pobreza;Clase media $16-$81 (2017 PPP);2010-2020;Redistribución;0
América Central;Brecha de pobreza;Pobreza $6.85 (2017 PPP);2010-2020;Total;-0,2
América Central;Brecha de pobreza;Pobreza $2.15 (2017 PPP);2010-2020;Crecimiento;-0,4
América Central;Brecha de pobreza;Pobreza $3.65 (2017 PPP);2010-2020;Crecimiento;-0,9
América Central;Brecha de pobreza;Clase media $16-$81 (2017 PPP);2010-2020;Total;-1
América Central;Brecha de pobreza;Clase media $16-$81 (2017 PPP);2010-2020;Crecimiento;-1
```

Fig63. PovertyMechanismsofchange.csv visualization in VSC

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

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
             País        Indicador             Línea de pobreza  Años Gap     Componente  Tasa
0     América Central  Brecha de pobreza  Clase media $16-$81 (2017 PPP)  2010-2015  Redistribución     0
1     América Central  Brecha de pobreza       Pobreza $2.15 (2017 PPP)  2010-2015  Redistribución  -0,1
2     América Central  Brecha de pobreza       Pobreza $3.65 (2017 PPP)  2010-2015  Redistribución  -0,3
3     América Central  Brecha de pobreza       Pobreza $2.15 (2017 PPP)  2010-2015     Crecimiento  -0,4
4     América Central  Brecha de pobreza       Pobreza $2.15 (2017 PPP)  2010-2015           Total  -0,5
...               ...               ...                            ...        ...             ...   ...
2515        Uruguay   Tasa de pobreza       Pobreza $2.15 (2017 PPP)  2015-2020  Redistribución  -0,1
2516        Uruguay   Tasa de pobreza       Pobreza $3.65 (2017 PPP)  2015-2020           Total  -0,3
2517        Uruguay   Tasa de pobreza       Pobreza $3.65 (2017 PPP)  2015-2020  Redistribución  -0,3
2518        Uruguay   Tasa de pobreza       Pobreza $6.85 (2017 PPP)  2015-2020           Total  -1,2
2519        Uruguay   Tasa de pobreza       Pobreza $6.85 (2017 PPP)  2015-2020  Redistribución  -1,2

[2520 rows x 6 columns]
(2520, 6)
```

Dataframe dimensions (2520 rows x 6 columns)

Fig64. Dataframe df5 visualization and df5 dimensions (2520 rows x 6 columns)

General data information

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

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2520 entries, 0 to 2519
Data columns (total 6 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   País             2520 non-null   object
 1   Indicador        2520 non-null   object
 2   Línea de pobreza  2520 non-null   object
 3   Años Gap         2520 non-null   object
 4   Componente       2520 non-null   object
 5   Tasa             2520 non-null   object
dtypes: object(6)
memory usage: 118.2+ KB
None
```

Fig65. df.info() attribute of df5 DataFrame

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

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Index(['País', 'Indicador', 'Línea de pobreza', 'Años Gap', 'Componente',
       'Tasa'],
      dtype='object')
```

Fig66. df.columns attribute of df5 DataFrame

Checking the data type

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

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
País              object
Indicador         object
Línea de pobreza  object
Años Gap          object
Componente        object
Tasa              object
dtype: object
```

Fig67. DataFrame df5 data types

Request a list of unique values

```
df5 = pd.read_csv( 'PovertyMechanismsofchange.csv', sep=";")
print(df5.Componente.unique())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
['Redistribución' 'Crecimiento' 'Total']
```

Fig68. unique() method

List of unique values and their frequency

```
df5 = pd.read_csv( 'PovertyMechanismsofchange.csv', sep=";")
print(df5.Componente.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Redistribución    840
Crecimiento       840
Total             840
Name: Componente, dtype: int64
```

Fig69. value_counts() method for component column

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

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
País             Indicador                Línea de pobreza                 Años Gap   Componente      Tasa
América Central  Brecha de pobreza        Clase media $16-$81 (2017 PPP)   2010-2015  Crecimiento     -1,3  1
México           Tasa de pobreza          Clase media $16-$81 (2017 PPP)   2015-2020  Crecimiento      0,1  1
                 Severidad de la pobreza  Vulnerable $6.85-$14 (2017 PPP)  2010-2015  Total           -0,2  1
                                                                           2015-2020  Crecimiento      0,7  1
                                                                                      Redistribución  -1,5  1
                                                                                                           ..
Colombia         Severidad de la pobreza  Pobreza $6.85 (2017 PPP)         2010-2020  Crecimiento      0,2  1
                                                                                      Redistribución   0,4  1
                                                                                      Total            0,5  1
                                                                           2015-2020  Crecimiento      1,8  1
Uruguay          Tasa de pobreza          Vulnerable $6.85-$14 (2017 PPP)  2015-2020  Total           -0,1  1
Length: 2520, dtype: int64
```

Fig70. value_counts() method for df5 DataFrame

Descriptive summary of the DataFrame df5

```
df5 = pd.read_csv( 'PovertyMechanismsofchange.csv', sep=";")
print(df5.describe())
print(df5.describe(include = object))
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
                País        Indicador            Línea de pobreza  Años Gap     Componente  Tasa
count           2520             2520                        2520      2520           2520  2520
unique            22                3                           5         3              3   205
top     América Central  Brecha de pobreza  Clase media $16-$81 (2017 PPP)  2010-2015  Redistribución     0
freq             135              840                         504       855            840   194
                País        Indicador            Línea de pobreza  Años Gap     Componente  Tasa
count           2520             2520                        2520      2520           2520  2520
unique            22                3                           5         3              3   205
top     América Central  Brecha de pobreza  Clase media $16-$81 (2017 PPP)  2010-2015  Redistribución     0
freq             135              840                         504       855            840   194
```

Fig71. Descriptive summary of the DataFrame df5

6.File_name:

PovertyRegionalDistribution.csv

Dataframe creation

We implement the creation of the df6 corresponding to the PovertyRegionalDistribution.csv file

PovertyIncomeContribution.csv

```
País;Tipo;Indicador;Línea de Pobreza;Componente;Años Gap;Tasa
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Ingresos laborales;2010-2020;1
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Ingresos laborales;2015-2020;0,8
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Porcentaje de empleados;2010-2020;0,7
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Porcentaje de empleados;2015-2020;0,6
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Total;2015-2020;0,4
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Total;2010-2020;0,3
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Porcentaje de empleados;2010-2015;0,3
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Ingresos laborales;2010-2015;0,3
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Jubilaciones y pensiones;2015-2020;0,1
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Remesas;2010-2020;0,1
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Porcentaje de individuos 15-64 años de edad;2010-2020;0,1
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Jubilaciones y pensiones;2010-2020;0,1
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Porcentaje de individuos 15-64 años de edad;2010-2015;0,1
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Remesas;2015-2020;0
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Porcentaje de individuos 15-64 años de edad;2015-2020;0
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Remesas;2010-2015;0
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Transferencias estatales;2010-2015;-0,1
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Total;2010-2015;-0,1
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Ingreso no laboral;2015-2020;-0,2
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Jubilaciones y pensiones;2010-2015;-0,3
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Ingreso no laboral;2010-2015;-0,4
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Ingreso no laboral;2010-2020;-0,6
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Transferencias estatales;2015-2020;-0,9
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $2.15 (2017 PPP);Transferencias estatales;2010-2020;-1
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $3.65 (2017 PPP);Ingresos laborales;2010-2020;1,7
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $3.65 (2017 PPP);Ingresos laborales;2015-2020;1,6
Argentina (urbano);Por fuentes de ingreso;Tasa de pobreza;Pobreza $3.65 (2017 PPP);Total;2015-2020;1,3
```

Fig72. PovertyRegionalDistribution.csv visualization in VSC

```
df6 = pd.read_csv( 'PovertyIncomeContribution.csv', sep=";")
print(df6)
print(df6.shape)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
              País              Tipo      Indicador       Línea de Pobreza                                  Componente  Años Gap Tasa
0     Argentina (urbano)  Por fuentes de ingreso  Tasa de pobreza  Pobreza $2.15 (2017 PPP)                                  Ingresos laborales  2010-2020   1
1     Argentina (urbano)  Por fuentes de ingreso  Tasa de pobreza  Pobreza $2.15 (2017 PPP)                                  Ingresos laborales  2015-2020  0,8
2     Argentina (urbano)  Por fuentes de ingreso  Tasa de pobreza  Pobreza $2.15 (2017 PPP)                               Porcentaje de empleados  2010-2020  0,7
3     Argentina (urbano)  Por fuentes de ingreso  Tasa de pobreza  Pobreza $2.15 (2017 PPP)                               Porcentaje de empleados  2015-2020  0,6
4     Argentina (urbano)  Por fuentes de ingreso  Tasa de pobreza  Pobreza $2.15 (2017 PPP)                                            Total  2015-2020  0,4
...                  ...                    ...              ...                      ...                                              ...        ...  ...
7784              Uruguay            Por género             Gini  Pobreza $6.85 (2017 PPP)                                            Total  2010-2015    0
7785              Uruguay            Por género             Gini  Pobreza $6.85 (2017 PPP)  Porcentaje de individuos 15-64 años de edad  2010-2015    0
7786              Uruguay            Por género             Gini  Pobreza $6.85 (2017 PPP)                               Porcentaje de empleados  2010-2015    0
7787              Uruguay            Por género             Gini  Pobreza $6.85 (2017 PPP)                                  Ingresos laborales  2010-2015    0
7788              Uruguay            Por género             Gini  Pobreza $6.85 (2017 PPP)                                  Ingreso no laboral  2010-2015    0

[7789 rows x 7 columns]
(7789, 7)
```

Dataframe dimensions (7789 rows x 7 columns)

Fig73. Dataframe df6 visualization and df4 dimensions (7789 rows x 7 columns)

General data information

```
df6 = pd.read_csv( 'PovertyIncomeContribution.csv', sep=";")
print(df6.info())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7789 entries, 0 to 7788
Data columns (total 7 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   País             7789 non-null   object
 1   Tipo             7789 non-null   object
 2   Indicador        7789 non-null   object
 3   Línea de Pobreza 7789 non-null   object
 4   Componente       7789 non-null   object
 5   Años Gap         7789 non-null   object
 6   Tasa             7789 non-null   object
dtypes: object(7)
memory usage: 426.1+ KB
None
```

Fig74. df.info() attribute of df6 DataFrame

```
df6 = pd.read_csv( 'PovertyIncomeContribution.csv', sep=";")
print(df6.columns)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Index(['País', 'Tipo', 'Indicador', 'Línea de Pobreza', 'Componente',
       'Años Gap', 'Tasa'],
      dtype='object')
```

Fig75. df.columns attribute of df6 DataFrame

Checking the data type

```
df6 = pd.read_csv( 'PovertyIncomeContribution.csv', sep=";")
print(df6.dtypes)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
País               object
Tipo               object
Indicador          object
Línea de Pobreza   object
Componente         object
Años Gap           object
Tasa               object
dtype: object
```

Fig76. DataFrame df6 data types

Request a list of unique values

```
df6 = pd.read_csv( 'PovertyIncomeContribution.csv', sep=";")
print(df6.Tasa.unique())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
['1' '0,8' '0,7' '0,6' '0,4' '0,3' '0,1' '0' '-0,1' '-0,2' '-0,3' '-0,4'
 '-0,6' '-0,9' '-1' '1,7' '1,6' '1,3' '1,1' '0,2' '-0,5' '-1,6' '-1,9'
 '5,3' '4,1' '4' '2,8' '1,5' '0,9' '-1,1' '-1,2' '-2,4' '-2,5' '0,5'
 '-0,7' '-0,8' '-1,3' '-1,4' '2,3' '-2' '-2,6' '-4,7' '3,6' '2,1' '1,4'
 '-2,8' '-2,9' '-6' '-2,2' '2' '-1,8' '-3,7' '-1,5' '-1,7' '-4,2' '2,5'
 '-2,1' '-3' '-3,5' '-6,4' '1,2' '-3,6' '-4,3' '-3,8' '2,2' '-3,2' '-4,5'
 '-4,8' '-3,1' '-3,3' '-2,7' '-3,4' '-3,9' '-4,9' '-2,3' '1,8' '1,9'
 '-4,1' '2,6' '-4,4' '-8,7' '-11' '-13,2' '-5,4' '3,2' '3' '-5,8' '-9,5'
 '2,4' '-6,1' '-8,6' '-15,5' '-8,3' '-5,5' '4,8' '4,4' '3,9' '3,1' '2,9'
 '6,3' '6,1' '4,3' '3,3' '-7' '7,6' '5,6' '4,7' '-10' '2,7' '3,8' '-5,7'
 '4,2' '3,5' '-5' '5' '-4,6' '-6,7' '4,5' '3,7' '-7,1' '-5,6' '5,8' '3,4'
 '-7,9' '-4' '-6,3' '-7,2' '-8,4' '-14,1' '-8,9' '-9,9' '-14' '-21,7'
 '-5,2' '-6,5' '-11,8' '-7,3' '-6,8' '6,8' '-9,8' '-15' '-5,3' '-9,6'
 '-5,1' '-11,4' '-11,7' '-5,9' '-6,2' '4,6' '10' '6,2' '5,2' '-7,8' '-8'
 '6' '-6,9' '-13,3']
```

Fig77. unique() method

List of unique values and their frequency

```
df6 = pd.read_csv( 'PovertyIncomeContribution.csv', sep=";")
print(df6.Tasa.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
0       2522
0,1      494
-0,1     494
0,2      349
-0,2     323
        ...
-5,5       1
3,7        1
6,1        1
4,5        1
-5,1       1
Name: Tasa, Length: 157, dtype: int64
```

Fig78. value_counts() method for Tasa column

```
df6 = pd.read_csv( 'PovertyIncomeContribution.csv', sep=";")
print(df6.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
País                Tipo                  Indicador          Línea de Pobreza         Componente                                 Años Gap   Tasa
Argentina (urbano)  Por fuentes de ingreso Brecha de pobreza  Pobreza $2.15 (2017 PPP) Ingreso no laboral                         2010-2015  -0,2    1
México              Por género             Tasa de pobreza    Pobreza $3.65 (2017 PPP) Porcentaje de individuos 15-64 años de edad 2015-2020  -0,6    1
Nicaragua           Por fuentes de ingreso Brecha de pobreza  Pobreza $3.65 (2017 PPP) Porcentaje de individuos 15-64 años de edad 2010-2020  -0,4    1
                                                                                       Porcentaje de empleados                    2010-2020  -1      1
                                                                                       Jubilaciones y pensiones                   2010-2020  -0,2    1
                                                                                                                                                    ..
Colombia            Por género             Tasa de pobreza    Pobreza $6.85 (2017 PPP) Ingresos laborales                         2010-2020  1,2     1
                                                                                                                                             -0,1    1
                                                                                                                                  2010-2015  -2,6    1
                                                                                                                                             -1,9    1
Uruguay             Por género             Tasa de pobreza    Pobreza $6.85 (2017 PPP) Total                                      2015-2020  -1,2    1
Length: 7789, dtype: int64
```

Fig79. value_counts() method for df6 DataFrame

Descriptive summary of the DataFrame df6

```
df6 = pd.read_csv( 'PovertyIncomeContribution.csv', sep=";")
print(df6.describe())
print(df6.describe(include = object))
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
                      País                    Tipo          Indicador          Línea de Pobreza          Componente  Años Gap  Tasa
count                 7789                    7789               7789                      7789                7789      7789  7789
unique                  18                       2                  4                         3                   8         3   157
top     Argentina (urbano)  Por fuentes de ingreso    Tasa de pobreza  Pobreza $3.65 (2017 PPP)  Ingresos laborales 2015-2020     0
freq                   522                    4320               2018                      2598                1475      2605  2522
                      País                    Tipo          Indicador          Línea de Pobreza          Componente  Años Gap  Tasa
count                 7789                    7789               7789                      7789                7789      7789  7789
unique                  18                       2                  4                         3                   8         3   157
top     Argentina (urbano)  Por fuentes de ingreso    Tasa de pobreza  Pobreza $3.65 (2017 PPP)  Ingresos laborales 2015-2020     0
freq                   522                    4320               2018                      2598                1475      2605  2522
```

Fig80. Descriptive summary of the DataFrame df6

7.File_name:

DesigualdadDistribuciónDeIngresos.csv

Dataframe creation

We implement the creation of the df6 corresponding to the PovertyRegionalDistribution.csv file

```
DesigualdadDistribuciónDeIngresos.csv
País;Año;Centiles;Proporción del ingreso;Distribución acumulada;Grupo
América Latina y el Caribe;2000;1;0,0;0;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;2;0,0;0;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;3;0,1;0;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;4;0,1;0;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;5;0,1;0;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;6;0,1;0;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;7;0,1;0;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;8;0,1;1;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;9;0,1;1;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;10;0,1;1;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;11;0,1;1;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;12;0,2;1;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;13;0,2;1;Pobreza $2.15 (2017 PPP)
América Latina y el Caribe;2000;14;0,2;1;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;15;0,2;2;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;16;0,2;2;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;17;0,2;2;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;18;0,2;2;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;19;0,2;2;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;20;0,2;3;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;21;0,2;3;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;22;0,2;3;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;23;0,3;3;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;24;0,3;4;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;25;0,3;4;Pobreza $3.65 (2017 PPP)
América Latina y el Caribe;2000;26;0,3;4;Pobreza $3.65 (2017 PPP)
```

Fig81. DesigualdadDistribuciónDeIngresos.csv visualization in VSC

```
df7 = pd.read_csv( 'DesigualdadDistribuciónDeIngresos.csv', sep=";")
print(df7)
print(df7.shape)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
                             País   Año  Centiles  Proporción del ingreso  Distribución acumulada                          Grupo
0      América Latina y el Caribe  2000         1                     0,0                       0       Pobreza $2.15 (2017 PPP)
1      América Latina y el Caribe  2000         2                     0,0                       0       Pobreza $2.15 (2017 PPP)
2      América Latina y el Caribe  2000         3                     0,1                       0       Pobreza $2.15 (2017 PPP)
3      América Latina y el Caribe  2000         4                     0,1                       0       Pobreza $2.15 (2017 PPP)
4      América Latina y el Caribe  2000         5                     0,1                       0       Pobreza $2.15 (2017 PPP)
...                           ...   ...       ...                     ...                     ...                            ...
30095                     Uruguay  2020        96                     2,6                      84  Clase media $16-$81 (2017 PPP)
30096                     Uruguay  2020        97                     2,9                      87  Clase media $16-$81 (2017 PPP)
30097                     Uruguay  2020        98                     3,3                      90                          Ricos
30098                     Uruguay  2020        99                     3,9                      94                          Ricos
30099                     Uruguay  2020       100                     6,2                     100                          Ricos

[30100 rows x 6 columns]
(30100, 6)
```

Dataframe dimensions (30100 rows x 6 columns)

Fig82. Dataframe df7 visualization and df4 dimensions (30100 rows x 6 columns)

General data information

```
df7 = pd.read_csv( 'DesigualdadDistribuciónDeIngresos.csv', sep=";")
print(df7.info())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30100 entries, 0 to 30099
Data columns (total 6 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   País                   30100 non-null  object
 1   Año                    30100 non-null  int64
 2   Centiles               30100 non-null  int64
 3   Proporción del ingreso 30100 non-null  object
 4   Distribución acumulada 30100 non-null  int64
 5   Grupo                  30100 non-null  object
dtypes: int64(3), object(3)
memory usage: 1.4+ MB
None
```

Fig83 df.info() attribute of df7 DataFrame

```
df7 = pd.read_csv( 'DesigualdadDistribuciónDeIngresos.csv', sep=";")
print(df7.columns)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
Index(['País', 'Año', 'Centiles', 'Proporción del ingreso',
       'Distribución acumulada', 'Grupo'],
      dtype='object')
```

Fig84. df.columns attribute of df7 DataFrame

Checking the data type

```
df7 = pd.read_csv( 'DesigualdadDistribuciónDeIngresos.csv', sep=";")
print(df7.dtypes)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
País                     object
Año                       int64
Centiles                  int64
Proporción del ingreso   object
Distribución acumulada    int64
Grupo                    object
dtype: object
```

Fig85. DataFrame df7 data types

Request a list of unique values

```
df7 = pd.read_csv( 'DesigualdadDistribuciónDeIngresos.csv', sep=";")
print(df7.Centiles.unique())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
[  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54
  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72
  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
  91  92  93  94  95  96  97  98  99 100]
```

Fig86. unique() method

List of unique values and their frequency

```
df7 = pd.read_csv( 'DesigualdadDistribuciónDeIngresos.csv', sep=";")
print(df7.Centiles.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
1      301
64     301
74     301
73     301
72     301
      ...
31     301
30     301
29     301
28     301
100    301
Name: Centiles, Length: 100, dtype: int64
```

Fig87. value_counts() method for Centiles column

```
df7 = pd.read_csv( 'DesigualdadDistribuciónDeIngresos.csv', sep=";")
print(df7.value_counts())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
País                     Año   Centiles  Proporción del ingreso  Distribución acumulada  Grupo
América Latina y el Caribe  2000  1         0,0                     0                       Pobreza $2.15 (2017 PPP)      1
Panamá                   2003  64        0,8                     24                      Clase media $16-$81 (2017 PPP)  1
                               76        1,2                     36                      Clase media $16-$81 (2017 PPP)  1
                               75        1,1                     34                      Clase media $16-$81 (2017 PPP)  1
                               74        1,1                     33                      Clase media $16-$81 (2017 PPP)  1
                                                                                                                    ..
Colombia                 2017  29        0,4                     7                       Pobreza $6.85 (2017 PPP)      1
                               28        0,4                     7                       Pobreza $6.85 (2017 PPP)      1
                               27        0,4                     7                       Pobreza $6.85 (2017 PPP)      1
                               26        0,4                     6                       Pobreza $6.85 (2017 PPP)      1
Uruguay                  2020  100       6,2                     100                     Ricos                         1
Length: 30100, dtype: int64
```

Fig88. value_counts() method for df7 DataFrame

```
df7 = pd.read_csv( 'DesigualdadDistribuciónDeIngresos.csv', sep=";")
print(df7.describe())
print(df7.describe(include = object))
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imp.py
                Año     Centiles  Distribución acumulada
count  30100.000000  30100.00000           30100.000000
mean    2010.385382     50.50000              25.945316
std        5.848485     28.86655              24.345706
min     2000.000000      1.00000               0.000000
25%     2005.000000     25.75000               6.000000
50%     2011.000000     50.50000              18.000000
75%     2015.000000     75.25000              40.000000
max     2020.000000    100.00000             100.000000
                          País Proporción del ingreso                    Grupo
count                    30100                30100                    30100
unique                      20                  147                        6
top     América Latina y el Caribe                  0,4  Clase media $16-$81 (2017 PPP)
freq                      2100                 2890                    10443
```

Fig89. Descriptive summary of the DataFrame df7

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

```
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    0
dtype: int64
```

0 data of type NaN

Fig90. pd.isnull() method

Next, it is confirmed that NaN type data is not presented by applying the notnull() method, this returns 115 notnull data, according to the previous stage the size of this Dataframe is (115, 16), therefore it is returning 115 records. In conclusion, the array does not present any data of type NaN.

```
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

Fig91. pd.notnull() method

Stage 6: Data preparation

In this stage all activities are implemented to build the data set to be used in the subsequent stage of modeling. Data preparation activities include data cleansing (dealing with invalid or missing values, remove duplicates and format properly), combine data from multiple sources (files, tables and platforms) and transform data into more useful variables.

The data preparation process for each set is presented below.

1. File_name: cityWorkCol.csv

Treatment of missing data

Entries missing values are given the value NaN, short for "Not a Number". For technical reasons these NaN values are always of the float64 dtype.

Pandas provide some specific methods for handling missing data. To select NaN entries, you can use pd.isnull() or its plugin pd.notnull(). These methods allow you to highlight values that are (or are not) empty (NaN).

It can be seen in the following figure that 0 data of the NaN type are presented in df1, that is, there are no data of the NaN type.

Data Type Conversion

As we observed in the preliminary analysis through the describe() method, we identified that only the Year column is of type int64 and the other columns are of type Object. To carry out the analysis corresponding to the DataFrame where it is required that these data be of a numeric type, such as the calculation of the mean, the variance, the quartiles, etc., a data type conversion must be implemented.

This conversion is done through the to_numeric() method, however, when applying this method, an error appears in the output, and two data cannot be transformed, as shown below:

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
df1["TGP"] = pd.to_numeric(df1["TGP"], downcast="float")

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning.py
Traceback (most recent call last):
  File "pandas\_libs\lib.pyx", line 2315, in pandas._libs.lib.maybe_convert_numeric
ValueError: Unable to parse string "61,13627832"

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "C:\Users\DELL\KAGGLE2FINALPRO\datacleaning.py", line 52, in <module>
    df1["TGP"] = pd.to_numeric(df1["TGP"], downcast="float")
  File "C:\Users\DELL\AppData\Local\Programs\Python\Python39\lib\site-packages\pandas\core\tools\numeric.py", line 184, in to_numeric
    values, _ = lib.maybe_convert_numeric(
  File "pandas\_libs\lib.pyx", line 2357, in pandas._libs.lib.maybe_convert_numeric
ValueError: Unable to parse string "61,13627832" at position 0
```

Fig92. Data transformation error

This type of error is very frequent, when data sets are obtained from external sources, most of the time these present inconsistencies, special characters within the data, and an endless number of situations that lead to the implementation of data-cleaning as a fundamental process in the data treatment.

To solve this error, the str.replace method is applied, which returns a copy of the string after replacing the occurrences of the old substring with new.

```
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
```

Fig93. to_numeric() and str.replace() methods

As can be seen in the previous image, all the data in the TPG column were changed from object type to float type, and even the mean() of the data was calculated as a final exercise.

df1.columns.str.strip() is applied which removes white space in column data. We proceed to change the data types of the other columns.

.

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
df1.columns = df1.columns.str.strip()
df1.rename(columns = {'%poblacion_en_edad_de_trabajar': 'Porcentaje_poblacion_en_edad_de_trabajar'})
df1['%poblacion_en_edad_de_trabajar'] = df1['%poblacion_en_edad_de_trabajar'].str.replace(',', '').astype(float)
df1['%poblacion_en_edad_de_trabajar'] = pd.to_numeric(df1['%poblacion_en_edad_de_trabajar'], downcast="float")

print(df1.columns)
print((df1.iloc[:, 3:4]).mean())
print(type(df1.iloc[:, 3:4]))

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning.py
Index(['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', 'Subocupados',
       'Fuerza_de_trabajo_potencial'],
      dtype='object')
%poblacion_en_edad_de_trabajar    6.069054e+09
dtype: float32
<class 'pandas.core.frame.DataFrame'>
```

Fig94. df1.columns.str.strip() method

Next, the corresponding columns to which you want to implement any numerical analysis will be transformed

```
df1 = pd.read_csv('cityWorkCol.csv', sep=";")
df1.columns = df1.columns.str.strip()

df1['TO'] = df1['TO'].str.replace(',', '').astype(float)
df1['TO'] = pd.to_numeric(df1['TO'], downcast="float")
print(df1.TO.dtype)

df1['TD'] = df1['TD'].str.replace(',', '').astype(float)
df1['TD'] = pd.to_numeric(df1['TD'], downcast="float")
print(df1.TD.dtype)

df1['TS'] = df1['TS'].str.replace(',', '').astype(float)
df1['TS'] = pd.to_numeric(df1['TS'], downcast="float")
print(df1.TS.dtype)

df1['Poblacion_total'] = df1['Poblacion_total'].str.replace(',', '').astype(float)
df1['Poblacion_total'] = pd.to_numeric(df1['Poblacion_total'], downcast="float")
print(df1.Poblacion_total.dtype)

df1['Poblacion_en_edad_de_trabajar'] = df1['Poblacion_en_edad_de_trabajar'].str.replace(',', '').astype(float)
df1['Poblacion_en_edad_de_trabajar'] = pd.to_numeric(df1['Poblacion_en_edad_de_trabajar'], downcast="float")
print(df1.Poblacion_en_edad_de_trabajar.dtype)

df1['Fuerza_de_trabajo'] = df1['Fuerza_de_trabajo'].str.replace(',', '').astype(float)
df1['Fuerza_de_trabajo'] = pd.to_numeric(df1['Fuerza_de_trabajo'], downcast="float")
print(df1.Fuerza_de_trabajo.dtype)

df1['Ocupados'] = df1['Ocupados'].str.replace(',', '').astype(float)
df1['Ocupados'] = pd.to_numeric(df1['Ocupados'], downcast="float")
print(df1.Ocupados.dtype)
```

Fig94. Data Type Conversion df1 Part1

```
df1['Desocupados'] = df1['Desocupados'].str.replace(',', '').astype(float)
df1['Desocupados'] = pd.to_numeric(df1['Desocupados'], downcast="float")
print(df1.Desocupados.dtype)


df1['Poblacion_fuera_de_la_fuerza_laboral'] = df1['Poblacion_fuera_de_la_fuerza_laboral'].str.replace(',', '').astype(float)
df1['Poblacion_fuera_de_la_fuerza_laboral'] = pd.to_numeric(df1['Poblacion_fuera_de_la_fuerza_laboral'], downcast="float")
print(df1.Poblacion_fuera_de_la_fuerza_laboral.dtype)

df1['Subocupados'] = df1['Subocupados'].str.replace(',', '').astype(float)
df1['Subocupados'] = pd.to_numeric(df1['Subocupados'], downcast="float")
print(df1.Subocupados.dtype)

df1['Fuerza_de_trabajo_potencial'] = df1['Fuerza_de_trabajo_potencial'].str.replace(',', '').astype(float)
df1['Fuerza_de_trabajo_potencial'] = pd.to_numeric(df1['Fuerza_de_trabajo_potencial'], downcast="float")
print(df1.Fuerza_de_trabajo_potencial.dtype)
```

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

Fig95. Data Type Conversion df1 Part2

As can be seen in the previous figure, they have been transformed to float 32 type, in this way it is possible to manipulate the data of this DataFrame for its respective analysis.

2. File_name: GEIH.csv

Treatment of missing data

As can be seen in the following images, zero NaN data records are obtained.

```
df1 = pd.read_csv('GEIH.csv', sep=";")
print(df2.isnull().sum())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning.py
ID_PERSONA,ID_HOGAR,ID_VIVIENDA,MCA_VIVIENDA_PROPIA,N_PERSONAS_HOGAR,TIPO_VIVIENDA,ESTRATO_VIVIENDA,NOMBRE_DEPTO,MCA_CONTRATO_ESCRITO,MCA_CONTRATO_TMNO_INDEFINIDO,MCA_BENEFICIOS_VACACIONES,MC
A_BENEFICIOS_PRIMA_NAVIDAD,MCA_BENEFICIOS_CESANTIAS,MCA_BENEFICIOS_AUX_ALIMENTACION,MCA_BENEFICIOS_SUB_TRANSPORTE,MCA_BENEFICIOS_SUB_FAMILIAR,MCA_BENEFICIOS_SUB_EDUCATIVO,MCA_BENEFICIOS_VIATI
COS_BONIFICACIONES,MCA_FONDO_PENSIONES,MCA_PENSIONADO,MCA_AFILIADO_ARL,MCA_CAJA_COMPENSACION,ANTIGUEDAD_MESES,HORAS_LABORALES_SEMANA,MCA_HR_EXTRA,MCA_SEGUNDO_TRABAJO,HORAS_SEGUNDO_TRABAJO_SEM
ANA,INGRESOS_HR_EXTRAS,INGRESOS_LABORALES,CATEGORIA_EMPLEADOS_EMPRESA,MCA_MUJER,MCA_JEFE_HOGAR,EDAD_ANOS,ESTADO_CIVIL,MCA_POBRE,NIVEL_EDUCATIVO,TITULO,ANOS_ESCOLARIDAD,MCA_ESTUDIA,MCA_ANALFAB
ETISMO,OI_ARRIENDOS,OI_CUOTA_ALIMENTARIA,OI_AYUDA_GOBIERNO,OI_AYUDA_EMPRIVADAS,OI_CESANTIAS,OI_FAMILIA_ACCION,OI_JOVENES_ACCION,OI_COLOMBIA_MAYOR,OI_OTRAS,INGRESOS_ADICIONALES,INGRESOS_FINALES
,CRIAR_ANIMALES,HORAS_SEM_CRIA_ANIM,OFICIOS_HOGAR,HORAS_SEM_OFIC_HOGAR,OFICIOS_OTROS,HORAS_SEM_OFIC_OTROS,CUIDAR_NINOS,HORAS_SEM_CUI_NINOS,CUIDAR_AMC_DISC,HORAS_SEM_CUIDAR_AMC_DIS,ELAB_PRENDA
S,HORAS_SEM_ELAB_PRENDAS,ASISTIR_EVENTOS,HORAS_SEM_ASIS_EVE,AUTOC_VIVIENDA,HORAS_SEM_AUT_VIVIENDA,TRAB_COMUNITARIO,HORAS_SEM_TRAB_CON_OTROS_TRAB_COMUNALES,HORAS_SEM_OTROS_COMUNALES,N_HIJOS,N
_HIJOS_MENORES_15ANOS,N_HIJOS_MENORES_10ANOS,MCA_CONYUGE_RESIDE  0
dtype: int64
```

Fig96. pd.isnull() method

```
df2 = pd.read_csv('GEIH.csv', sep=";")
print(df2.notnull().sum())
```

```
ID_PERSONA,ID_HOGAR,ID_VIVIENDA,MCA_VIVIENDA_PROPIA,N_PERSONAS_HOGAR,TIPO_VIVIENDA,ESTRATO_VIVIENDA,NOMBRE_DEPTO,MCA_CONTRATO_ESCRITO,MCA_CONTRATO_TMNO_INDEFINIDO,MCA_BENEFICIOS_VACACIONES,MC
A_BENEFICIOS_PRIMA_NAVIDAD,MCA_BENEFICIOS_CESANTIAS,MCA_BENEFICIOS_AUX_ALIMENTACION,MCA_BENEFICIOS_SUB_TRANSPORTE,MCA_BENEFICIOS_SUB_FAMILIAR,MCA_BENEFICIOS_SUB_EDUCATIVO,MCA_BENEFICIOS_VIATI
COS_BONIFICACIONES,MCA_FONDO_PENSIONES,MCA_PENSIONADO,MCA_AFILIADO_ARL,MCA_CAJA_COMPENSACION,ANTIGUEDAD_MESES,HORAS_LABORALES_SEMANA,MCA_HR_EXTRA,MCA_SEGUNDO_TRABAJO,HORAS_SEGUNDO_TRABAJO_SEM
ANA,INGRESOS_HR_EXTRAS,INGRESOS_LABORALES,CATEGORIA_EMPLEADOS_EMPRESA,MCA_MUJER,MCA_JEFE_HOGAR,EDAD_ANOS,ESTADO_CIVIL,MCA_POBRE,NIVEL_EDUCATIVO,TITULO,ANOS_ESCOLARIDAD,MCA_ESTUDIA,MCA_ANALFAB
ETISMO,OI_ARRIENDOS,OI_CUOTA_ALIMENTARIA,OI_AYUDA_GOBIERNO,OI_AYUDA_EMPRIVADAS,OI_CESANTIAS,OI_FAMILIA_ACCION,OI_JOVENES_ACCION,OI_COLOMBIA_MAYOR,OI_OTRAS,INGRESOS_ADICIONALES,INGRESOS_FINALES
,CRIAR_ANIMALES,HORAS_SEM_CRIA_ANIM,OFICIOS_HOGAR,HORAS_SEM_OFIC_HOGAR,OFICIOS_OTROS,HORAS_SEM_OFIC_OTROS,CUIDAR_NINOS,HORAS_SEM_CUI_NINOS,CUIDAR_AMC_DISC,HORAS_SEM_CUIDAR_AMC_DIS,ELAB_PRENDA
S,HORAS_SEM_ELAB_PRENDAS,ASISTIR_EVENTOS,HORAS_SEM_ASIS_EVE,AUTOC_VIVIENDA,HORAS_SEM_AUT_VIVIENDA,TRAB_COMUNITARIO,HORAS_SEM_TRAB_CON_OTROS_TRAB_COMUNALES,HORAS_SEM_OTROS_COMUNALES,N_HIJOS,N
_HIJOS_MENORES_15ANOS,N_HIJOS_MENORES_10ANOS,MCA_CONYUGE_RESIDE  28634
dtype: int64
```

Fig97. pd.notnull() method

Data Type Conversion

In the process of visualizing the data types, it was observed that they are all of the object types, therefore the respective conversion of the type is carried out for further analysis.

However, when trying to make the corresponding transformations, the following error appears:

```
df2 = pd.read_csv('GEIH.csv', sep=";")
df2.columns = df2.columns.str.strip()
df2['N_PERSONAS_HOGAR'] = df2['N_PERSONAS_HOGAR'].str.replace(',', '').astype(float)
df2['N_PERSONAS_HOGAR'] = pd.to_numeric(df2['N_PERSONAS_HOGAR'], downcast="float")
print(df2.N_PERSONAS_HOGAR.dtype)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning.py
Traceback (most recent call last):
  File "C:\Users\DELL\AppData\Local\Programs\Python\Python39\lib\site-packages\pandas\core\indexes\base.py", line 3621, in get_loc
    return self._engine.get_loc(casted_key)
  File "pandas\_libs\index.pyx", line 136, in pandas._libs.index.IndexEngine.get_loc
  File "pandas\_libs\index.pyx", line 163, in pandas._libs.index.IndexEngine.get_loc
  File "pandas\_libs\hashtable_class_helper.pxi", line 5198, in pandas._libs.hashtable.PyObjectHashTable.get_item
  File "pandas\_libs\hashtable_class_helper.pxi", line 5206, in pandas._libs.hashtable.PyObjectHashTable.get_item
KeyError: 'N_PERSONAS_HOGAR'
```

Fig98. Data transformation error

In this way, the error is overcome and the transformation of the data type is carried out successfully.

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode')
df2.columns = df2.columns.str.strip()
df2['N_PERSONAS_HOGAR'] = df2['N_PERSONAS_HOGAR'].str.replace(',', '').astype(float)
df2['N_PERSONAS_HOGAR'] = pd.to_numeric(df2['N_PERSONAS_HOGAR'], downcast="float")
print(df2.N_PERSONAS_HOGAR.dtype)
```

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

Fig99Application dtype='unicode'

By applying dtype=unicode, the data transformation can be implemented via the astype (float) method. The following shows the data transformation of the other columns of the DataFrame.

```
df2 [ "TIPO_VIVIENDA" ]  =  df2 [ "TIPO_VIVIENDA" ] . astype ( float )
print(df2.TIPO_VIVIENDA.dtype)

df2 [ "ESTRATO_VIVIENDA" ]  =  df2 [ "ESTRATO_VIVIENDA" ] . astype ( float )
print(df2.ESTRATO_VIVIENDA.dtype)

df2 [ "MCA_CONTRATO_ESCRITO" ]  =  df2 [ "MCA_CONTRATO_ESCRITO" ] . astype ( float )
print(df2.MCA_CONTRATO_ESCRITO.dtype)

df2 [ "MCA_CONTRATO_TMNO_INDEFINIDO" ]  =  df2 [ "MCA_CONTRATO_TMNO_INDEFINIDO" ] . astype ( float )
print(df2.MCA_CONTRATO_TMNO_INDEFINIDO.dtype)

df2 [ "MCA_BENEFICIOS_VACACIONES" ]  =  df2 [ "MCA_BENEFICIOS_VACACIONES" ] . astype ( float )
print(df2.MCA_BENEFICIOS_VACACIONES.dtype)

df2 [ "MCA_BENEFICIOS_PRIMA_NAVIDAD" ]  =  df2 [ "MCA_BENEFICIOS_PRIMA_NAVIDAD" ] . astype ( float )
print(df2.MCA_BENEFICIOS_PRIMA_NAVIDAD.dtype)

df2 [ "MCA_BENEFICIOS_CESANTIAS" ]  =  df2 [ "MCA_BENEFICIOS_CESANTIAS" ] . astype ( float )
print(df2.MCA_BENEFICIOS_CESANTIAS.dtype)

df2 [ "MCA_BENEFICIOS_AUX_ALIMENTACION" ]  =  df2 [ "MCA_BENEFICIOS_AUX_ALIMENTACION" ] . astype ( float )
print(df2.MCA_BENEFICIOS_AUX_ALIMENTACION.dtype)
```

```
df2 [ "MCA_BENEFICIOS_SUB_FAMILIAR" ]  =  df2 [ "MCA_BENEFICIOS_SUB_FAMILIAR" ] . astype ( float )
print(df2.MCA_BENEFICIOS_SUB_FAMILIAR.dtype)

df2 [ "MCA_BENEFICIOS_SUB_EDUCATIVO" ]  =  df2 [ "MCA_BENEFICIOS_SUB_EDUCATIVO" ] . astype ( float )
print(df2.MCA_BENEFICIOS_SUB_EDUCATIVO.dtype)

df2 [ "MCA_FONDO_PENSIONES" ]  =  df2 [ "MCA_FONDO_PENSIONES" ] . astype ( float )
print(df2.MCA_FONDO_PENSIONES.dtype)

df2 [ "MCA_PENSIONADO" ]  =  df2 [ "MCA_PENSIONADO" ] . astype ( float )
print(df2.MCA_PENSIONADO.dtype)

df2 [ "MCA_AFILIADO_ARL" ]  =  df2 [ "MCA_AFILIADO_ARL" ] . astype ( float )
print(df2.MCA_AFILIADO_ARL.dtype)

df2 [ "MCA_CAJA_COMPENSACION" ]  =  df2 [ "MCA_CAJA_COMPENSACION" ] . astype ( float )
print(df2.MCA_CAJA_COMPENSACION.dtype)

df2 [ "MCA_AFILIADO_ARL" ]  =  df2 [ "MCA_AFILIADO_ARL" ] . astype ( float )
print(df2.MCA_AFILIADO_ARL.dtype)

df2 [ "ANTIGUEDAD_MESES" ]  =  df2 [ "ANTIGUEDAD_MESES" ] . astype ( float )
print(df2.ANTIGUEDAD_MESES.dtype)

df2 [ "HORAS_LABORALES_SEMANA" ]  =  df2 [ "HORAS_LABORALES_SEMANA" ] . astype ( float )
print(df2.HORAS_LABORALES_SEMANA.dtype)
```

```
df2 [ "MCA_HR_EXTRA" ]  =  df2 [ "MCA_HR_EXTRA" ] . astype ( float )
print(df2.MCA_HR_EXTRA.dtype)

df2 [ "MCA_SEGUNDO_TRABAJO" ]  =  df2 [ "MCA_SEGUNDO_TRABAJO" ] . astype ( float )
print(df2.MCA_SEGUNDO_TRABAJO.dtype)

df2 [ "HORAS_SEGUNDO_TRABAJO_SEMANA" ]  =  df2 [ "HORAS_SEGUNDO_TRABAJO_SEMANA" ] . astype ( float )
print(df2.HORAS_SEGUNDO_TRABAJO_SEMANA.dtype)

df2 [ "INGRESOS_HR_EXTRAS" ]  =  df2 [ "INGRESOS_HR_EXTRAS" ] . astype ( float )
print(df2.INGRESOS_HR_EXTRAS.dtype)

df2 [ "INGRESOS_LABORALES" ]  =  df2 [ "INGRESOS_LABORALES" ] . astype ( float )
print(df2.INGRESOS_LABORALES.dtype)

df2 [ "CATEGORIA_EMPLEADOS_EMPRESA" ]  =  df2 [ "CATEGORIA_EMPLEADOS_EMPRESA" ] . astype ( float )
print(df2.CATEGORIA_EMPLEADOS_EMPRESA.dtype)

df2 [ "MCA_MUJER" ]  =  df2 [ "MCA_MUJER" ] . astype ( float )
print(df2.MCA_MUJER.dtype)

df2 [ "MCA_JEFE_HOGAR" ]  =  df2 ["MCA_MUJER"] . astype ( float )
print(df2.MCA_MUJER.dtype)

df2 [ "EDAD_ANOS" ]  =  df2 [ "EDAD_ANOS" ] . astype ( float )
print(df2.EDAD_ANOS.dtype)
```

```
df2 [ "ESTADO_CIVIL" ]  =  df2 [ "ESTADO_CIVIL" ] . astype ( float )
print(df2.ESTADO_CIVIL.dtype)

df2 [ "MCA_POBRE" ]  =  df2 [ "MCA_POBRE" ] . astype ( float )
print(df2.MCA_POBRE.dtype)

df2 [ "NIVEL_EDUCATIVO" ]  =  df2 [ "NIVEL_EDUCATIVO" ] . astype ( float )
print(df2.NIVEL_EDUCATIVO.dtype)

df2 [ "TITULO" ]  =  df2 [ "TITULO" ] . astype ( float )
print(df2.TITULO.dtype)

df2 [ "ANOS_ESCOLARIDAD" ]  =  df2 [ "ANOS_ESCOLARIDAD" ] . astype ( float )
print(df2.ANOS_ESCOLARIDAD.dtype)

df2 [ "MCA_ESTUDIA" ]  =  df2 [ "MCA_ESTUDIA" ] . astype ( float )
print(df2.MCA_ESTUDIA.dtype)

df2 [ "MCA_ANALFABETISMO" ]  =  df2 [ "MCA_ANALFABETISMO" ] . astype ( float )
print(df2.MCA_ANALFABETISMO.dtype)

df2 [ "OI_ARRIENDOS" ]  =  df2 [ "OI_ARRIENDOS" ] . astype ( float )
print(df2.OI_ARRIENDOS.dtype)

df2 [ "OI_CUOTA_ALIMENTARIA" ]  =  df2 [ "OI_CUOTA_ALIMENTARIA" ] . astype ( float )
print(df2.OI_CUOTA_ALIMENTARIA.dtype)
```

```
df2 [ "OI_AYUDA_GOBIERNO" ]  =  df2 [ "OI_AYUDA_GOBIERNO" ] . astype ( float )
print(df2.OI_AYUDA_GOBIERNO.dtype)

df2 [ "OI_AYUDA_EPRIVADAS" ]  =  df2 [ "OI_AYUDA_EPRIVADAS" ] . astype ( float )
print(df2.OI_AYUDA_EPRIVADAS.dtype)

df2 [ "OI_CESANTIAS" ]  =  df2 [ "OI_CESANTIAS" ] . astype ( float )
print(df2.OI_CESANTIAS.dtype)

df2 [ "OI_FAMILIA_ACCION" ]  =  df2 [ "OI_FAMILIA_ACCION" ] . astype ( float )
print(df2.OI_FAMILIA_ACCION.dtype)

df2 [ "OI_JOVENES_ACCION" ]  =  df2 [ "OI_JOVENES_ACCION" ] . astype ( float )
print(df2.OI_JOVENES_ACCION.dtype)

df2 [ "OI_COLOMBIA_MAYOR" ]  =  df2 [ "OI_COLOMBIA_MAYOR" ] . astype ( float )
print(df2.OI_COLOMBIA_MAYOR.dtype)

df2 [ "OI_OTRAS" ]  =  df2 [ "OI_OTRAS" ] . astype ( float )
print(df2.OI_OTRAS.dtype)

df2 [ "INGRESOS_ADICIONALES" ]  =  df2 [ "INGRESOS_ADICIONALES" ] . astype ( float )
print(df2.INGRESOS_ADICIONALES.dtype)
```

```
df2 [ "INGRESOS_FINALES" ]  =  df2 [ "INGRESOS_FINALES" ] . astype ( float )
print(df2.INGRESOS_FINALES.dtype)

df2 [ "CRIAR_ANIMALES" ]  =  df2 [ "CRIAR_ANIMALES" ] . astype ( float )
print(df2.CRIAR_ANIMALES.dtype)

df2 [ "HORAS_SEM_CRIA_ANIM" ]  =  df2 [ "HORAS_SEM_CRIA_ANIM" ] . astype ( float )
print(df2.HORAS_SEM_CRIA_ANIM.dtype)

df2 [ "OFICIOS_HOGAR" ]  =  df2 [ "OFICIOS_HOGAR" ] . astype ( float )
print(df2.OFICIOS_HOGAR.dtype)

df2 [ "HORAS_SEM_OFIC_HOGAR" ]  =  df2 [ "HORAS_SEM_OFIC_HOGAR" ] . astype ( float )
print(df2.HORAS_SEM_OFIC_HOGAR.dtype)

df2 [ "OFICIOS_OTROS" ]  =  df2 [ "OFICIOS_OTROS" ] . astype ( float )
print(df2.OFICIOS_OTROS.dtype)

df2 [ "HORAS_SEM_OFIC_OTROS" ]  =  df2 [ "HORAS_SEM_OFIC_OTROS" ] . astype ( float )
print(df2.HORAS_SEM_OFIC_OTROS.dtype)

df2 [ "CUIDAR_NINOS" ]  =  df2 [ "CUIDAR_NINOS" ] . astype ( float )
print(df2.CUIDAR_NINOS.dtype)

df2 [ "HORAS_SEM_CUI_NINOS" ]  =  df2 [ "HORAS_SEM_CUI_NINOS" ] . astype ( float )
print(df2.HORAS_SEM_CUI_NINOS.dtype)
```

```
df2 [ "CUIDAR_ANC_DISC" ] = df2 [ "CUIDAR_ANC_DISC" ] . astype ( float )
print(df2.CUIDAR_ANC_DISC.dtype)

df2 [ "HORAS_SEM_CUIDAR_ANC_DIS" ] = df2 [ "HORAS_SEM_CUIDAR_ANC_DIS" ] . astype ( float )
print(df2.HORAS_SEM_CUIDAR_ANC_DIS.dtype)

df2 [ "AUTOC_VIVIENDA" ] = df2 [ "AUTOC_VIVIENDA" ] . astype ( float )
print(df2.HORAS_SEM_CUIDAR_ANC_DIS.dtype)

df2 [ "HORAS_SEM_AUT_VIVIENDA" ] = df2 [ "HORAS_SEM_AUT_VIVIENDA" ] . astype ( float )
print(df2.HORAS_SEM_AUT_VIVIENDA.dtype)

df2 [ "TRAB_COMUNITARIO" ] = df2 [ "TRAB_COMUNITARIO" ] . astype ( float )
print(df2.TRAB_COMUNITARIO.dtype)

df2 [ "HORAS_SEM_TRAB_COM" ] = df2 [ "HORAS_SEM_TRAB_COM" ] . astype ( float )
print(df2.HORAS_SEM_TRAB_COM.dtype)

df2 [ "N_HIJOS" ] = df2 [ "N_HIJOS" ] . astype ( float )
print(df2.N_HIJOS.dtype)
```

```
df2 [ "N_HIJOS_MENORES_15ANOS" ] = df2 [ "N_HIJOS_MENORES_15ANOS" ] . astype ( float )
print(df2.N_HIJOS_MENORES_15ANOS.dtype)

df2 [ "N_HIJOS_MENORES_10ANOS" ] = df2 [ "N_HIJOS_MENORES_10ANOS" ] . astype ( float )
print(df2.N_HIJOS_MENORES_10ANOS.dtype)

df2 [ "MCA_CONYUGE_RESIDE" ] = df2 [ "MCA_CONYUGE_RESIDE" ] . astype ( float )
print(df2.MCA_CONYUGE_RESIDE.dtype)
```

Fig100. Aplication astype (float) method in df2

Duplicate records

Next we check if there are duplicate records:

```
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
```

Fig101. Duplicate record analysis

As can be seen in the image above, there are no duplicate records in the DataFrame df2

3. File_name: PovertyIncidenceRate.csv

Treatment of missing data

Parsing the amount of NaN data via the pd.isnull() method returns the following:

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3.isnull().sum())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning2.py
Area            0
País            0
Línea Pobreza   0
Indicador       0
Año             0
Tasa          253
dtype: int64
```

Fig102. pd.isnull() method

There are 253 records with NaN value in the Tasa column. The first step to take is to analyze the data set paying special attention to the Tasa column.

PovertyIncidenceRate.csv

```
Area;País;Línea Pobreza;Indicador;Año;Tasa
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2003;13,2
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2004;12,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2002;12,2
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2000;12
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2001;11,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2005;11,6
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2006;10,4
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2020;10,2
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2007;8,7
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2008;8,6
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2009;8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2012;7,9
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2010;7,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2011;7,6
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2018;7
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2013;6,9
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2019;6,8
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2016;6,7
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2017;6,6
Nacional;América Central;Pobreza $2.15 (2017 PPP);Tasa de pobreza;2014;6,6
```

Fig103. PovertyIncidenceRate.csv visualization in VSC

The Tasa column corresponds to the value given as a percentage of the poverty rate both for the region in general and for the countries that comprise it.

Through the isnull() method we trace the records corresponding to the Rate column that present data of type NaN

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
print(df3.Tasa.isnull)

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning2.py
<bound method Series.isnull of 0        13,2
1        12,8
2        12,2
3          12
4        11,8
         ...
10621     NaN
10622     NaN
10623     NaN
10624     NaN
10625     NaN
Name: Tasa, Length: 10626, dtype: object>
```

Fig104. pd.isnull() method applied to Tasa column

To identify the Dataframe records that contain NaN values and to be able to better analyze the steps to follow, the variable: nan_rows = df3[df3.isnull().any(1)] is created, which will contain all the records with NaN values in the array.

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
nan_rows = df3[df3.isnull().any(1)]
print(nan_rows)

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning2.py
       Area              País                 Línea Pobreza              Indicador   Año Tasa
7546   Rural  Argentina (urbano)       Pobreza $2.15 (2017 PPP)  Severidad de la pobreza  2020  NaN
7547   Rural  Argentina (urbano)       Pobreza $2.15 (2017 PPP)        Brecha de pobreza  2020  NaN
7548   Rural  Argentina (urbano)       Pobreza $2.15 (2017 PPP)          Tasa de pobreza  2020  NaN
7549   Rural  Argentina (urbano)       Pobreza $2.15 (2017 PPP)  Severidad de la pobreza  2019  NaN
7550   Rural  Argentina (urbano)       Pobreza $2.15 (2017 PPP)        Brecha de pobreza  2019  NaN
...      ...                ...                            ...                       ...   ...  ...
10621  Rural            Uruguay  Clase media $14-$81 (2017 PPP)          Tasa de pobreza  2004  NaN
10622  Rural            Uruguay  Clase media $14-$81 (2017 PPP)          Tasa de pobreza  2003  NaN
10623  Rural            Uruguay  Clase media $14-$81 (2017 PPP)          Tasa de pobreza  2002  NaN
10624  Rural            Uruguay  Clase media $14-$81 (2017 PPP)          Tasa de pobreza  2001  NaN
10625  Rural            Uruguay  Clase media $14-$81 (2017 PPP)          Tasa de pobreza  2000  NaN

[253 rows x 6 columns]
```

Fig105. Dataframe records that contain NaN values

As can be seen in the previous image, 253 records are obtained that contain NAN values in the Tasa column of the DataFrame.

The first step to consider when obtaining this type of data is to try to communicate with the dataset collection source, in this case the World Bank. In this case, this step is not possible due to time constraints, we proceed to analyze it in depth of the previously obtained image where other relevant data for further analysis are observed; as the poverty line and its respective indicator.

Due to the previous finding it is decided:

In the first place, do not apply the replace() method, since by entering data randomly the statistical analyzes will be altered and since it is a poverty indicator, one must be much more prudent in its manipulation.

Secondly, do not apply the dropna() method that allows you to filter the data by returning a DataFrame without the records that contain NaN values. This decision is made based on the other data found in said records, described above, which may alter other important measurements to be analyzed.

Finally, the decision is made to apply the fillna() method, through which each NaN with an "Unknown" will be replaced:

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";")
unknown = df3.Tasa.fillna("Unknown")
print(unknown)

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning2.py
0          13,2
1          12,8
2          12,2
3            12
4          11,8
           ...
10621    Unknown
10622    Unknown
10623    Unknown
10624    Unknown
10625    Unknown
Name: Tasa, Length: 10626, dtype: object
```

Fig106. fillna() method applied to Tasa column

The previous figure shows how the NAN data types were changed by the value "Unknown".

Data Type Conversion

In the process of visualizing the data types, it was observed that they are all of the object types, therefore the respective conversion of the type is carried out for further analysis.

It is important to transform the Tasa column to a numeric type in case you need to carry out mathematical or statistical operations with the percentage of the poverty index.

The process is shown below:

```
df3 = pd.read_csv('PovertyIncidenceRate.csv', sep=";", dtype='unicode')
df3.columns = df3.columns.str.strip()
df3['Tasa'] = df3['Tasa'].str.replace(',', '').astype(float)
df3['Tasa'] = pd.to_numeric(df3['Tasa'], downcast="float")
print(df3.Tasa.dtype)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning2.py
float32
```

Fig107. Aplication pd.to_numeric() method in df3

4.File_name: PovertyIncomeContribution.csv

It can be seen in the following figure that 0 data of the NaN type are presented in df4, that is, there are no data of the NaN type.

```
df4 = pd.read_csv('PovertyIncomeContribution.csv', sep=";")
print(df4.isnull().sum())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning2.py
País              0
Tipo              0
Indicador         0
Línea de Pobreza  0
Componente        0
Años Gap          0
Tasa              0
dtype: int64
```

Fig110. pd.isnull() method

In the process of visualizing the data types, it was observed that they are all of the object types, therefore the respective conversion of the type is carried out for further analysis.

It is important to transform the Tasa column to a numeric type in case you need to carry out mathematical or statistical operations with the percentage of the poverty index.

The process is shown below:

```
df4 = pd.read_csv('PovertyIncidenceRate.csv', sep=";", dtype='unicode')
df4.columns = df4.columns.str.strip()
df4['Tasa'] = df4['Tasa'].str.replace(',', '').astype(float)
df4['Tasa'] = pd.to_numeric(df4['Tasa'], downcast="float")
print(df4.Tasa.dtype)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning2.py
float32
```

Fig111. Aplication pd.to_numeric() method in df4

5.File_name:

PovertyMechanismsofchange.csv

It can be seen in the following figure that 0 data of the NaN type are presented in df5, that is, there are no data of the NaN type

```
df5 = pd.read_csv('PovertyMechanismsofchange.csv', sep=";")
print(df5.isnull().sum())
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning2.py
País              0
Indicador         0
Línea de pobreza  0
Años Gap          0
Componente        0
Tasa              0
dtype: int64
```

Fig112. pd.isnull() method

However, when analyzing the csv file, it is observed that the Poverty Line column contains 5 different data per record.

PovertyMechanismsofchange.csv

```
1  País;Indicador;Línea de pobreza;Años Gap;Componente;Tasa
2  América Central;Brecha de pobreza;Clase media $16-$81 (2017 PPP);2010-2015;Redistribución;0
3  América Central;Brecha de pobreza;Pobreza $2.15 (2017 PPP);2010-2015;Redistribución;-0,1
4  América Central;Brecha de pobreza;Pobreza $3.65 (2017 PPP);2010-2015;Redistribución;-0,3
5  América Central;Brecha de pobreza;Pobreza $2.15 (2017 PPP);2010-2015;Crecimiento;-0,4
6  América Central;Brecha de pobreza;Pobreza $2.15 (2017 PPP);2010-2015;Total;-0,5
7  América Central;Brecha de pobreza;Vulnerable $6.85-$14 (2017 PPP);2010-2015;Redistribución;-0,5
8  América Central;Brecha de pobreza;Pobreza $6.85 (2017 PPP);2010-2015;Redistribución;-0,7
9  América Central;Brecha de pobreza;Pobreza $3.65 (2017 PPP);2010-2015;Crecimiento;-1,1
10 América Central;Brecha de pobreza;Clase media $16-$81 (2017 PPP);2010-2015;Total;-1,2
11 América Central;Brecha de pobreza;Clase media $16-$81 (2017 PPP);2010-2015;Crecimiento;-1,3
12 América Central;Brecha de pobreza;Pobreza $3.65 (2017 PPP);2010-2015;Total;-1,4
13 América Central;Brecha de pobreza;Pobreza $6.85 (2017 PPP);2010-2015;Crecimiento;-2,2
14 América Central;Brecha de pobreza;Pobreza $6.85 (2017 PPP);2010-2015;Total;-2,9
15 América Central;Brecha de pobreza;Vulnerable $6.85-$14 (2017 PPP);2010-2015;Crecimiento;-3
```

Fig113.Linea de Pobreza column

These data should be separated and placed in individual columns for further analysis. The new column labels are:

- Classification: Contains the nominal classification of the socio-economic situation.

- IngresoMin: Corresponds to the minimum value of the range established within the classification column
- IngresoMax: Corresponds to the maximum value of the range established within the classification column
- Año_Medicion: Corresponds to the year in which the different measurements were made, for the entire Dataframe it is 2017.
- PurPoPa: Contains Purchasing power parities (PPPs) are the rates of currency conversion that try to equalize the purchasing power of different currencies, by eliminating the differences in price levels between countries.

This process is done through the concat() merge method. as shown below:

```
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)
```

```
            País        Indicador            Línea de pobreza  Años Gap     Componente  ... Clasificaion IngresoMin IngresoMax Año_Medicion PurPo
Pa
0     América Central  Brecha de pobreza  Clase media $16-$81 (2017 PPP)  2010-2015  Redistribución  ...      Clase     media     $16-$81        (2017   PP
P)
1     América Central  Brecha de pobreza       Pobreza $2.15 (2017 PPP)  2010-2015  Redistribución  ...    Pobreza     $2.15      (2017         PPP)   No
ne
2     América Central  Brecha de pobreza       Pobreza $3.65 (2017 PPP)  2010-2015  Redistribución  ...    Pobreza     $3.65      (2017         PPP)   No
ne
3     América Central  Brecha de pobreza       Pobreza $2.15 (2017 PPP)  2010-2015     Crecimiento  ...    Pobreza     $2.15      (2017         PPP)   No
ne
4     América Central  Brecha de pobreza       Pobreza $2.15 (2017 PPP)  2010-2015           Total  ...    Pobreza     $2.15      (2017         PPP)   No
ne
...               ...               ...                            ...        ...             ...  ...        ...       ...        ...          ...   .
..
2515        Uruguay  Tasa de pobreza       Pobreza $2.15 (2017 PPP)  2015-2020  Redistribución  ...    Pobreza     $2.15      (2017         PPP)   No
ne
2516        Uruguay  Tasa de pobreza       Pobreza $3.65 (2017 PPP)  2015-2020           Total  ...    Pobreza     $3.65      (2017         PPP)   No
ne
2517        Uruguay  Tasa de pobreza       Pobreza $3.65 (2017 PPP)  2015-2020  Redistribución  ...    Pobreza     $3.65      (2017         PPP)   No
ne
2518        Uruguay  Tasa de pobreza       Pobreza $6.85 (2017 PPP)  2015-2020           Total  ...    Pobreza     $6.85      (2017         PPP)   No
ne
2519        Uruguay  Tasa de pobreza       Pobreza $6.85 (2017 PPP)  2015-2020  Redistribución  ...    Pobreza     $6.85      (2017         PPP)   No
ne

[2520 rows x 11 columns]
```

Fig114. Aplication concat() method in df5

In the previous stage, it was determined that the dimensions of the DataFrame were: (2520 rows x 6 columns), as can be seen in the previous image, the new dimensions of the DataFrame are (2520 rows x 11 columns), this is due to the separation of data made previously.

Data Type Conversion

In the process of visualizing the data types, it was observed that they are all of the object types, therefore the respective conversion of the type is carried out for further analysis.

It is important to transform the Tasa, column to a numeric type in case you need to carry out mathematical or statistical operations with the percentage of the poverty index.

First, the $ sign must be removed from the records belonging to the columns IngresoMin and IngresoMax. This process is done by implementing the replace() method m through a λ function.

```
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)
```

```
            País        Indicador            Línea de pobreza  Años Gap     Componente  ... Clasificaion IngresoMin IngresoMax Año_Medicion PurPo
Pa
0     América Central  Brecha de pobreza  Clase media $16-$81 (2017 PPP)  2010-2015  Redistribución  ...      Clase     media      16-81        (2017   PP
P)
1     América Central  Brecha de pobreza       Pobreza $2.15 (2017 PPP)  2010-2015  Redistribución  ...    Pobreza      2.15      (2017         PPP)   No
ne
2     América Central  Brecha de pobreza       Pobreza $3.65 (2017 PPP)  2010-2015  Redistribución  ...    Pobreza      3.65      (2017         PPP)   No
ne
3     América Central  Brecha de pobreza       Pobreza $2.15 (2017 PPP)  2010-2015     Crecimiento  ...    Pobreza      2.15      (2017         PPP)   No
ne
4     América Central  Brecha de pobreza       Pobreza $2.15 (2017 PPP)  2010-2015           Total  ...    Pobreza      2.15      (2017         PPP)   No
ne
...               ...               ...                            ...        ...             ...  ...        ...       ...        ...          ...   .
..
2515        Uruguay  Tasa de pobreza       Pobreza $2.15 (2017 PPP)  2015-2020  Redistribución  ...    Pobreza      2.15      (2017         PPP)   No
ne
2516        Uruguay  Tasa de pobreza       Pobreza $3.65 (2017 PPP)  2015-2020           Total  ...    Pobreza      3.65      (2017         PPP)   No
ne
2517        Uruguay  Tasa de pobreza       Pobreza $3.65 (2017 PPP)  2015-2020  Redistribución  ...    Pobreza      3.65      (2017         PPP)   No
ne
2518        Uruguay  Tasa de pobreza       Pobreza $6.85 (2017 PPP)  2015-2020           Total  ...    Pobreza      6.85      (2017         PPP)   No
ne
2519        Uruguay  Tasa de pobreza       Pobreza $6.85 (2017 PPP)  2015-2020  Redistribución  ...    Pobreza      6.85      (2017         PPP)   No
ne

[2520 rows x 11 columns]
```

Fig115. Elimination of the symbol $ from the columns IngresoMin and IngresoMax

```
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)

df['Tasa'] = df['Tasa'].str.replace(',', '').astype(float)
df['Tasa'] = pd.to_numeric(df['Tasa'], downcast="float")
print(df.Tasa.dtype)

[2520 rows x 11 columns]
float32
```

Fig116. Aplication pd.to_numeric() method in df5

6.File_name:

PovertyRegionalDistribution.csv

It can be seen in the following figure that 0 data of the NaN type are presented in df6, that is, there are no data of the NaN type

```
df3 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
print(df3.isnull().sum())

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning3.py
Tipo              0
País              0
Year              0
Línea de Pobreza  0
Share             0
dtype: int64
```

Fig117. pd.isnull() method

Data Type Conversion

In the process of visualizing the data types, it was observed that they are all of the object types, therefore the respective conversion of the type is carried out for further analysis.

It is important to transform the Share column to a numeric type in case you need to carry out mathematical or statistical operations with the percentage of the poverty index.

To carry out this process, the symbols ($), (%) must be eliminated from the data in the Share column, for this the replace() method is applied through λ functions.

The process is shown below:

```
df6 = pd.read_csv('PovertyRegionalDistribution.csv', sep=";")
df6.columns = df6.columns.str.strip()

df6["Share"] = df6["Share"].apply(lambda x: x.replace("$",""))
df6["Share"] = df6["Share"].apply(lambda y: y.replace("%",""))
print(df6)

df6['Share'] = df6['Share'].str.replace(',', '').astype(float)
df6['Share'] = pd.to_numeric(df6['Share'], downcast="float")
print(df6.Share.dtype)

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning3.py
            Tipo               País  Year          Línea de Pobreza Share
0        Por país  Argentina (urbano)  2002  Pobreza $6.85 (2017 PPP)   3,3
1        Por país  Argentina (urbano)  2003  Pobreza $6.85 (2017 PPP)   3,1
2        Por país  Argentina (urbano)  2002  Pobreza $3.65 (2017 PPP)   2,9
3        Por país  Argentina (urbano)  2001  Pobreza $6.85 (2017 PPP)   2,8
4        Por país  Argentina (urbano)  2002  Pobreza $2.15 (2017 PPP)   2,7
...           ...                ...   ...                       ...   ...
1381  Sub-regional       Región Andina  2014  Pobreza $6.85 (2017 PPP)  21,3
1382  Sub-regional       Región Andina  2018  Pobreza $3.65 (2017 PPP)  21,1
1383  Sub-regional       Región Andina  2017  Pobreza $2.15 (2017 PPP)  20,2
1384  Sub-regional       Región Andina  2018  Pobreza $2.15 (2017 PPP)  18,7
1385  Sub-regional       Región Andina  2019  Pobreza $2.15 (2017 PPP)  18,6

[1386 rows x 5 columns]
float32
```

Fig118. Elimination of the symbols ($), (%) from the column Share

7.File_name:

DesigualdadDistribuciónDeIngresos.csv

It can be seen in the following figure that 0 data of the NaN type are presented in df6, that is, there are no data of the NaN type

```
df7 = pd.read_csv('DesigualdadDistribuciónDeIngresos.csv', sep=";")
print(df7.isnull().sum())

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\datacleaning3.py
País                      0
Año                       0
Centiles                  0
Proporción del ingreso    0
Distribución acumulada    0
Grupo                     0
dtype: int64
```

Fig119. pd.isnull() method

Data Type Conversion

In the process of visualizing the data types, it was observed that they are all of the object types, therefore the respective conversion of the type is carried out for further analysis.

It is important to transform the Centiles column to a numeric type in case you need to carry out mathematical or statistical operations with the percentage of the poverty index.

The process is shown below:

```
df7 = pd.read_csv('DesigualdadDistribuciónDeIngresos.csv', sep=";")
df7.columns = df7.columns.str.strip()
df7['Centiles'] = pd.to_numeric(df7['Centiles'], downcast="float")
print(df7.Centiles.dtype)

float32
```

Fig120. Data type conversion in the Centiles column.

Stage 7: Modeling

In this stage, the generation of descriptive models on the sets obtained in the previous stage is defined in the first instance.

Secondly, the generation of predictive models is defined through the application of ML techniques.

Stage 8: Evaluation and Implementation

A. 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.

An equitable society is defined as one in which people have the same opportunities to follow the life they choose, regardless of the circumstances into which they were born, and are not subject to poverty (World Bank, 2005).

In this order of ideas, we proceed to analyze the socio-economic situation of Colombians according to the selected data sets.

We now analyze the socio-economic stratum variable, identified with the label (ESTRATO_VIVIENDA), as a fundamental element for targeting the population in a state of vulnerability in the five main cities of the country (Bogotá, Medellin, Cali, Barranquilla, and Bucaramanga).

First, a Series is created through the groupby() method that allows grouping all the values corresponding to the socio-economic stratum variable present in the dataset. The count() method is also applied, which generates the frequency of occurrence.

The following image shows the process mentioned previously

```
df2 = pd.read_csv('GEIH.csv' , dtype='unicode')
group_estrato = df2.groupby('ESTRATO_VIVIENDA').ESTRATO_VIVIENDA.count()
print(group_estrato)

C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imple1.py
ESTRATO_VIVIENDA
0      15
1    3283
2    7305
3    6611
4    2104
5     737
6     463
9      16
```

Fig121. Frecuency of Socio-Economic Stratum

In order to achieve a better analysis of the process obtained above, the series is plotted using the seaborn tool as shown below:

```
df2 = pd.read_csv('GEIH.csv' , dtype='unicode')
sns.countplot(x =df2.ESTRATO_VIVIENDA, palette = sns.color_palette("pastel" ),
              saturation = 1 ). set (title = 'Frequency of socio-economic stratum ')
plt.show()
```

Fig122: Frequency of socio-economic stratum code

This tool allows the behavior of the count() method to be evidenced through a bar graph that relates the analyzed variable with the respective frequency.

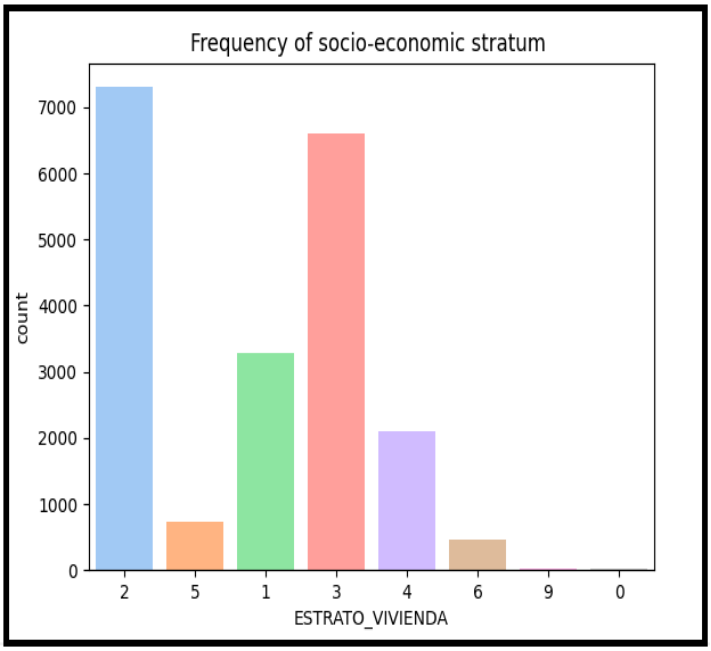


Fig123. Frequency of Socio-Economic Stratum

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.

In order to determine the respective percentages corresponding to each of the strata, the following procedure is implemented:

The group_estrato series created in the previous step is converted to the group_estrato1 DataFrame.

It is relevant to note that stratum 9, the original value of the dataset, does not correspond to a socio-economic stratification level. By methodology of the GEIH 2021, this value corresponds to the citizens who answered Don't know/No answer in this question (Annex 2). For this reason, the respective label for stratum 9 is changed to Ns/Nr.

The sort_values() method is applied to sort the DataFrame with respect to the column previously created for strata.

The implementation of the process explained above is shown below:

```
df2 = pd.read_csv('GEIH.csv' , dtype='unicode')
group_estrato = df2.groupby('ESTRATO_VIVIENDA').ESTRATO_VIVIENDA.count()
print(group_estrato)
group_estrato1 = group_estrato.to_frame()
print(group_estrato1)
print(group_estrato1.columns)
group_estrato1["Estratos"] = ["Estrato_0", "Estrato_1", "Estrato_2", "Estrato_3","Estrato_4",
                              "Estrato_5", "Estrato_6", "Ns/Nr"]
by_estratos = group_estrato1.sort_values('Estratos')
print(group_estrato1)

import plotly.express as px

fig = px.pie(group_estrato1, values = "ESTRATO_VIVIENDA", color = "Estratos", names = "Estratos",
             title = 'Frequency of Socio-Sconomic Stratum ',
             color_discrete_map = {'Estrato_0': '#FFB6C1',
                                   'Estrato_1': '#8690FF',
                                   'Estrato_2': '#30BFDD',
                                   'Estrato_3': '#FF00FF',
                                   'Estrato_4': '#00FFFF',
                                   'Estrato_5': '#FFFACD',
                                   'Estrato_6': '#FFB9B3',
                                   'Ns/Nr': '#1E90FF'

                                   }
            )

fig.show()
```

ESTRATO_VIVIENDA	ESTRATO_VIVIENDA	Estratos
0	15	Estrato_0
1	3283	Estrato_1
2	7305	Estrato_2
3	6611	Estrato_3
4	2104	Estrato_4
5	737	Estrato_5
6	463	Estrato_6
9	16	Ns/Nr

Fig124. al DataFrame group_estrato1

Finally, we proceed to implement a pie chart of the information generated, using the Plotly tool. In this way, the percentages corresponding to each of the socio-economic strata are obtained.

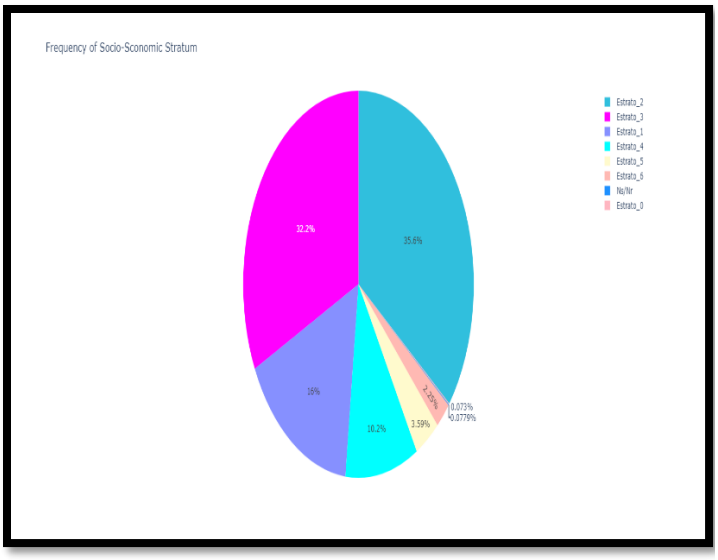


Fig125. Percentages of Socio-Economic Strata.

With respect to the previous graph it can be concluded that 51.6% of the population is in the low classification, that is to say in poverty.

Stratum 3 represents 32.2% of the population and is classified as lower-medium. It is mostly made up of people from lower strata who have achieved social mobility through their own efforts, generally through academic training at a technological level or in public institutions of higher education, as well as through the creation of small commercial establishments (trade between neighborhoods).

This population group is in great danger of being relegated to socio-economic strata 1 and 2 because they do not have savings or the necessary resources to assume an eventuality or fortuitous event that may occur.

An example of this situation is the pandemic; "the middle class contracted from 14.7 million people in 2019 to 12.5 million in 2020, which implies that 2.5 million Colombians dropped out of this category, due to a deterioration in their income... "The most important impact we are seeing is on the fraction of the population that is middle class. While the country observed 30.1% of its population in the middle class by the 2019 income distribution, we are seeing a reduction in this case of 4.6 percentage points in the weight of this group in the country" DANE (2020).

This implies that, during 2020, the population belonging to the middle class of the country, i.e. stratum 3, was the hardest hit, generating 3.5 million new poor people in just one year, going from 17.4 million to 21.0 million people, generating a setback in social mobility.

The fact that the vulnerable population is falling back into poverty has devastating consequences for the national economy. For the IDB, "the middle class can move the national economy and generate prosperity for more people because of its impact on the generation of capital, its purchasing power, its demand for quality goods and its multiplier effect on spending (chaining)".

Continuing with the analysis, stratum 4 corresponds to 10.2% of the Colombian population and is classified as Medium concerning its purchasing power.

Stratum 5 corresponds to 3.59% of the population and is classified as medium-high concerning its purchasing power.

Only 2.25% of the entire Colombian population is classified as stratum 6 which corresponds to a high level of purchasing power.

By combining the population corresponding to strata 5 and 6, we obtain that only 5.84% of the citizens of the 5 largest cities in Colombia have high purchasing power.

The above analysis corroborates the serious situation of inequality in Colombia. "Income inequality in Colombia is very high. In 2019, it was the highest among all OECD countries, and most Latin American and Caribbean (LAC) countries. Moreover, inequality in Colombia has been on the rise since 2018 and was further exacerbated by the impact of COVID-19. The Gini coefficient of household income (a standard measure of inequality) reached 0.53 in 2019, after taxes and transfers" (World Bank, 2021).

Similarly, the UNDP warned in 2020 that the "economic recovery, like the crisis, will not be the same for everyone; the most vulnerable people will have more difficulties".

A survey by the Inter-American Development Bank (IDB) revealed that while 80 % of lower-income families suffered job losses, this situation was experienced by only 15 to 20 % of higher-income families. One of the most powerful reasons for this difference is that informality and underemployment are largely found in strata 1, 2, and 3, while the other strata have formal jobs, which allowed them to telecommute during the period of confinement.

It is important to contrast at this point the figures given by DANE (2020) with those obtained in this analysis. DANE reports 30.1% of the population in the middle class, i.e. stratum 3, and the graph under analysis reports 32.2% of the Colombian population in this stratum.

This is an indicator that the sample taken from the GEIH 2021 corresponding to the main cities up to this point represents the study population.

We proceed to analyze the variable NAME_DEPTO, which contains the names of the five main cities in the country where the GEIH has been carried out, corresponding to Bogotá, Medellin, Cali, Barranquilla, and Bucaramanga.

Applying the procedure, first of all, a Series is created through the groupby() method that allows grouping all the values corresponding to the socio-economic stratum variable present in the dataset.

Likewise, the count() method is applied, which generates the frequency of occurrence.

The following graph shows the aforementioned process:

```
group_by_Nombre_Depto = df2.groupby('NOMBRE_DEPTO').NOMBRE_DEPTO.count()
print(group_by_Nombre_Depto)
```

```
NOMBRE_DEPTO
B/manga         3037
Barranquilla    3482
Bogota          4740
Cali            2693
Medellin        6582
Name: NOMBRE_DEPTO, dtype: int64
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imple1.py
NOMBRE_DEPTO
B/manga         3037
Barranquilla    3482
Bogota          4740
Cali            2693
Medellin        6582
Name: NOMBRE_DEPTO, dtype: int64
```

Fig125: Cities frequency code

In order to achieve a better analysis of the process obtained above, the Series is plotted using the Seaborn tool, as shown below:

```
sns.countplot(x =df2.NOMBRE_DEPTO, palette = sns.color_palette("pastel" ),
              saturation = 1 ). set (title = 'Frequency of socio-economic stratum ')
plt.show()
```

Fig127. Cities frecuency graph code

The following graph presents the frequency of the cities:

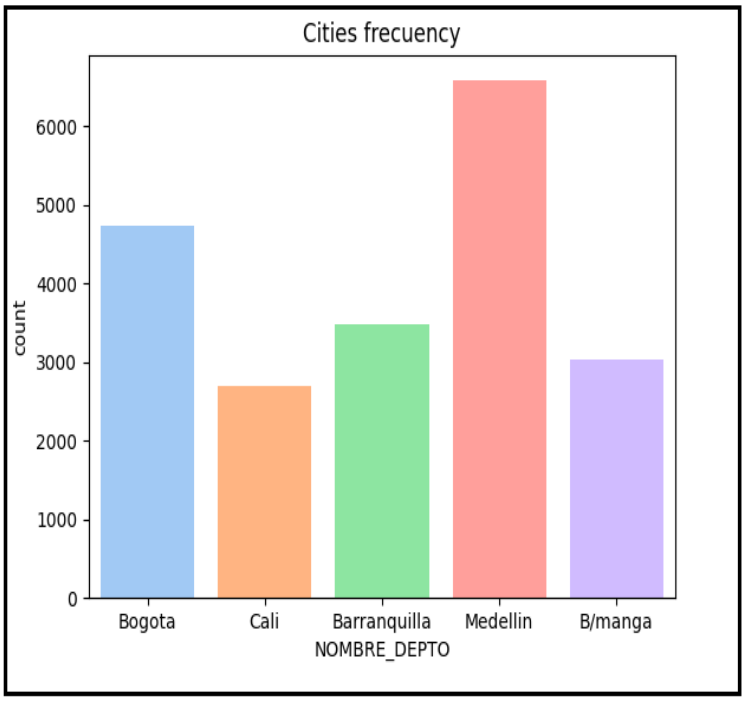


Fig126. Cities frecuency

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.

According to data from DANE (2020) one year before this survey was implemented, the cities had the following number of inhabitants:

City	Population
Medellín	2569007
Cali	2496346
Bogotá	8380801
Barranquilla	1239804
Bucaramanga	528 572

Table7. The population of the five main cities

Next, the series generated in the previous step is converted to the: DataFramegroup_by_Department_Name1.

Subsequently, the Population column is created, which contains the data corresponding to the population of each of the cities analyzed.

The cities column is also created, which contains the data corresponding to the cities analyzed.

Finally, the sort_values() method whose parameter is Population is applied. This creates the Dataframe by_population, which contains all the implementations described above.

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

group_by_Nombre_Depto = df2.groupby('NOMBRE_DEPTO').NOMBRE_DEPTO.count()
print(group_by_Nombre_Depto)
group_by_Nombre_Depto1 = group_by_Nombre_Depto.to_frame()
print(group_by_Nombre_Depto1)

group_by_Nombre_Depto1["Population"] = ["528572", "1239804", "8380801", "2496346", "2569007"]
print(group_by_Nombre_Depto1)

group_by_Nombre_Depto1["cities"] = ["B/manga", "Barranquilla", "Bogota", "Cali", "Medellin"]
print(group_by_Nombre_Depto1)

by_population = group_by_Nombre_Depto1.sort_values('Population')
print(by_population)
```

Fig127a DataFrame by_population code Part1

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\imple1.py
NOMBRE_DEPTO
B/manga         3037
Barranquilla    3482
Bogota          4740
Cali            2693
Medellin        6582
Name: NOMBRE_DEPTO, dtype: int64
              NOMBRE_DEPTO
NOMBRE_DEPTO
B/manga               3037
Barranquilla          3482
Bogota                4740
Cali                  2693
Medellin              6582
              NOMBRE_DEPTO Population
NOMBRE_DEPTO
B/manga               3037     528572
Barranquilla          3482    1239804
Bogota                4740    8380801
Cali                  2693    2496346
Medellin              6582    2569007
              NOMBRE_DEPTO Population        cities
NOMBRE_DEPTO
B/manga               3037     528572       B/manga
Barranquilla          3482    1239804  Barranquilla
Bogota                4740    8380801        Bogota
Cali                  2693    2496346          Cali
Medellin              6582    2569007       Medellin
              NOMBRE_DEPTO Population        cities
NOMBRE_DEPTO
Barranquilla          3482    1239804  Barranquilla
Cali                  2693    2496346          Cali
Medellin              6582    2569007       Medellin
B/manga               3037     528572       B/manga
Bogota                4740    8380801        Bogota
```

Fig127b DataFrame by_population code Part2

Finally, we proceed to implement a pie chart of the information generated, using the Plotly tool. In this way, the percentages corresponding to each of the cities in which the survey was implemented are obtained.

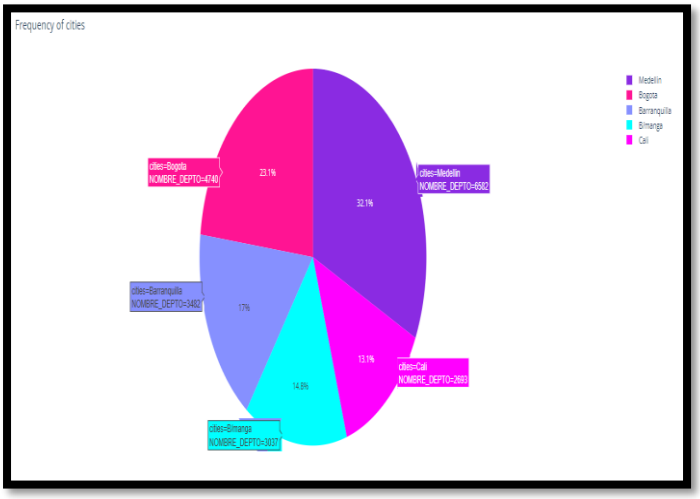


Fig128. Percentage cities

According to the graph above, Medellin has a 32.1% share in the GEIH, Bogotá 23.1%, Barranquilla 17%, Bucaramanga 14.8% and Cali with a 13.1 share.

Next, we proceed to graph the percentage of the population interviewed by cities.

```
fig = px.pie(group_by_Nombre_Depto1, values = "Population", color = "cities", names = "cities",
             title = 'Population percentage ',
             color_discrete_map = {'Bogota':  '#FF1493',
                                   'Barranquilla': '#8690FF',
                                   'Medellin': '#8A2BE2',
                                   'Cali': '#FF00FF',
                                   'B/manga': '#00FFFF',



                                   }
    )

fig.show()
```

Fig129 Percentage of population interviewed by city code

Below is the image corresponding to the previous code.

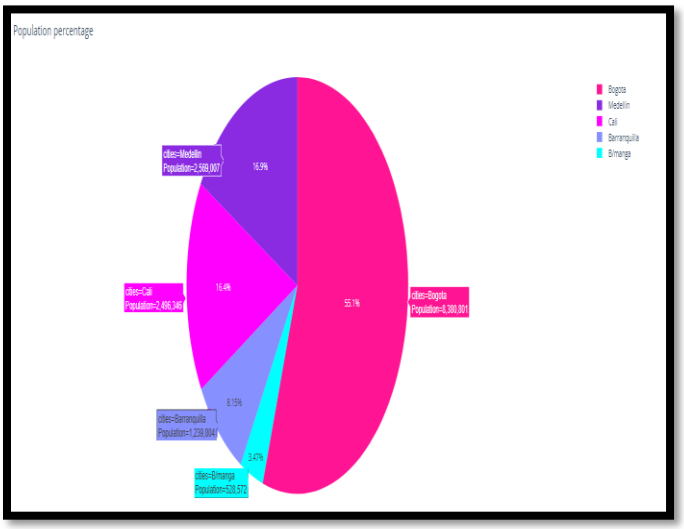


Fig130 Population percentage

For the target population of the five main cities in Colombia, the population distribution is distributed as follows: Bogotá concentrates 55.1% of the population distribution, Medellín 16.9%, Cali 16.4%, Barranquilla 8.15% and Bucaramanga 3.47%.

Analysis of the variable Years of education

This variable corresponds to the years of formal academic education a citizen has completed. The variable takes values from 0 to 26 and is represented by the label YEARS_EDUCATION.

We proceed to implement its analysis. First, it determines the respective frequency of each of the elements of the variable through the groupby () method and proceeds to plot it through Seaborn's countplot tool. The code is shown below:

```
df2 = pd.read_csv('GEIH.csv', index_col=0)
group_by_ANOS_ESCOLARIDAD = df2.groupby('ANOS_ESCOLARIDAD').ANOS_ESCOLARIDAD.count()
print(group_by_ANOS_ESCOLARIDAD)

sns.countplot(x =df2.ANOS_ESCOLARIDAD, palette = sns.color_palette("pastel" ),
              saturation = 1 ). set (title = 'Frecuency years of education')
plt.show()
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\groupby.py
ANOS_ESCOLARIDAD
0.0       27
1.0       22
2.0       49
3.0       99
4.0       86
5.0      654
6.0      155
7.0      223
8.0      256
9.0      327
10.0     141
11.0    6913
12.0     494
13.0    2459
14.0    1779
15.0     601
16.0    3434
17.0     821
18.0     948
19.0     354
20.0     149
21.0     131
22.0      44
23.0      42
24.0      13
25.0       5
26.0      23
Name: ANOS_ESCOLARIDAD, dtype: int64
```

Fig131 Frequency of the variable Years of Education code.

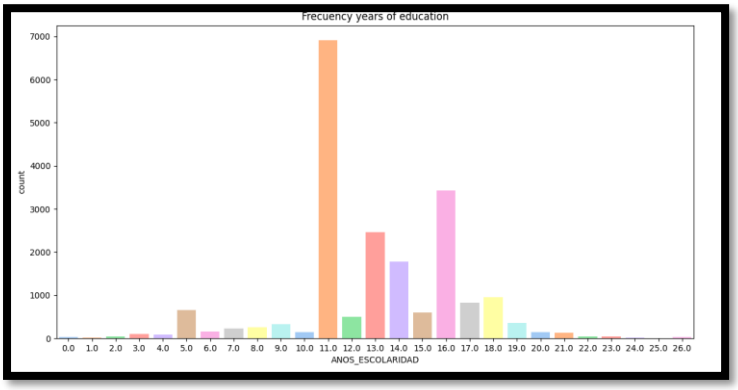


Fig132: Frequency of the variable Years of Education code.

The following is an analysis of the stages of academic education in Colombia:

Early childhood education consists of three (3) years of education at the pre-kindergarten, kindergarten, and transition levels. It is important to bear in mind for the corresponding analysis that the transition grade, which is aimed at five (5) year old students, corresponds to the constitutional compulsory grade (Decree 2247 of September 11, 1997). In other words, all Colombians must have completed at least the transition grade to continue their academic education in basic primary education.

Basic Primary: corresponding to five (5) years of education.

Basic secondary education: corresponding to four (4) years of education.

Secondary education: two (2) years of education corresponding to grades 10 and 11.

Taking into account the above, a Colombian needs a minimum of ten (10) years of education to obtain a basic secondary education.

He/she would also need two (2) more years, for a total of twelve (12) years of schooling to obtain the baccalaureate degree corresponding to the secondary education level.

If one wishes to become a technician, one (1) more year of academic training is required for thirteen (13) years.

To opt for the title of technologist, a citizen is required to complete the training corresponding to the secondary education level plus two (2) years of education, i.e. a total of fourteen (14) years.

To be eligible for an undergraduate degree, a citizen must have a high school education plus five (5) years of university preparation, for a total of seventeen (17) years. For a total of 17 years.

From 17 years of education onwards, the undergraduate stage is presented, considering that the student has only completed the transition grade in his/her early childhood education stage.

In this way, a citizen who has completed all of his or her early childhood education will require twelve (12) years to complete basic secondary education, fourteen (14) years to complete secondary education, fifteen (15) years to complete technical education, sixteen (16) years to complete technological education, and nineteen (19) years to complete the undergraduate stage, i.e. to obtain a professional degree.

Training of more than nineteen (19) years in Colombia corresponds to the post-graduate stage.

In this order of ideas, 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.

Low-frequency values are also observed for years of professional training corresponding to the undergraduate level, with the visualization at the postgraduate level being more critical.

The above analysis is corroborated by the study "Probability of obtaining an academic degree in Colombia" developed by Cerquera-Losada, O., Gómez-Segura, C. and Rojas-Velásquez, L. (2021). In which it is analyzed, according to figures from the Gran Encuesta Integrada de Hogares (GEIH) for the year 2018, of the total Colombian population of working age 54% did not attain any academic degree, only 3 out of 10 have a bachelor's degree, 8.2% are technicians or technologists and 5.9% are university graduates. According to Ferreyra (2017), the gaps in access to university education are because the vast majority of students in Colombia come from low-income sectors. They do not manage to graduate from high school or if they do graduate, their academic level is below average, which limits the possibility of achieving a university education degree (Cerquera, Gómez, Rojas, 2021).

Next, the percentage corresponding to the frequency of each value of the variable will be determined by implementing the following procedure:

The series generated in the previous process is transformed to the DataFrame group_by_YEARS_SCHOOL1.

Then a new column is created with the name Years, containing all the labels corresponding to the variable.

Finally, the sort_values() method is applied whose argument will be the previously created Years column.

The procedure is shown below;

```
df2 = pd.read_csv('GEIH.csv', index_col=0)
group_by_ANOS_ESCOLARIDAD = df2.groupby('ANOS_ESCOLARIDAD').ANOS_ESCOLARIDAD.count()
print(group_by_ANOS_ESCOLARIDAD)
group_by_ANOS_ESCOLARIDAD1 = group_by_ANOS_ESCOLARIDAD.to_frame()
group_by_ANOS_ESCOLARIDAD1["years"] = ["0Y", "1Y", "2Y", "3y", "4Y", "5Y", "6Y", "7Y","8Y","9Y","10Y","11Y",
                                       "12Y", "13Y", "14Y", "15Y", "16Y", "17Y", "18Y", "19Y", "20Y",
                                       "21Y", "22Y", "23Y", "24Y", "25Y", "26Y" ]
group_by_ANOS_ESCOLARIDAD1 = group_by_ANOS_ESCOLARIDAD1.sort_values('years')
print(group_by_ANOS_ESCOLARIDAD1)
```

ANOS_ESCOLARIDAD	ANOS_ESCOLARIDAD	years
0.0	27	0Y
10.0	141	10Y
11.0	6913	11Y
12.0	494	12Y
13.0	2459	13Y
14.0	1779	14Y
15.0	601	15Y
16.0	3434	16Y
17.0	821	17Y
18.0	948	18Y
19.0	354	19Y
1.0	22	1Y
20.0	149	20Y
21.0	131	21Y
22.0	44	22Y
23.0	42	23Y
24.0	13	24Y
25.0	5	25Y
26.0	23	26Y
2.0	49	2Y
3.0	99	3y
4.0	86	4Y
5.0	654	5Y
6.0	155	6Y
7.0	223	7Y
8.0	256	8Y
9.0	327	9Y

Fig133: Percentage of the variable Years of Education code.

The development of the corresponding graph is presented below.

```
df2 = pd.read_csv('GEIH.csv', index_col=0)
group_by_ANOS_ESCOLARIDAD = df2.groupby('ANOS_ESCOLARIDAD').ANOS_ESCOLARIDAD.count()
print(group_by_ANOS_ESCOLARIDAD)
group_by_ANOS_ESCOLARIDAD1 = group_by_ANOS_ESCOLARIDAD.to_frame()
group_by_ANOS_ESCOLARIDAD1["years"] = ["0Y", "1Y", "2Y", "3y", "4Y", "5Y", "6Y", "7Y","8Y","9Y","10Y","11Y",
                                       "12Y", "13Y", "14Y", "15Y", "16Y", "17Y", "18Y", "19Y", "20Y",
                                       "21Y", "22Y", "23Y", "24Y", "25Y", "26Y" ]
group_by_ANOS_ESCOLARIDAD1 = group_by_ANOS_ESCOLARIDAD1.sort_values('years')
print(group_by_ANOS_ESCOLARIDAD1)

fig = px.pie(group_by_ANOS_ESCOLARIDAD1, values = "ANOS_ESCOLARIDAD", color = "years", names = "years",
             title = 'Percentage years of education ')

fig.show()
```

Fig134: Percentage graph of the variable Years of Education code.

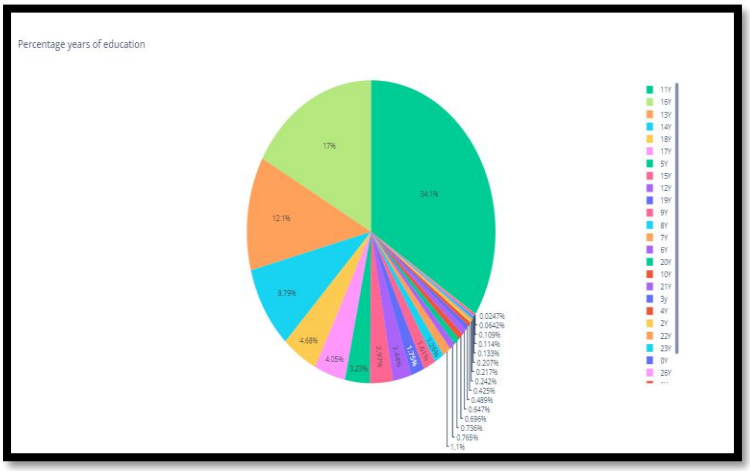


Fig135: Percentage graph of the variable Years of Education.

It can be seen that the highest percentage 34.1% corresponds to citizens who have completed 11 years of academic education, i.e. third of the population has a basic secondary education.

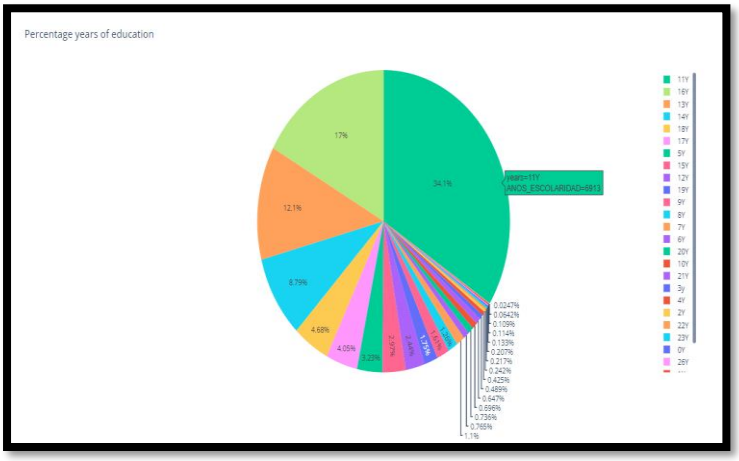


Fig136: Graph Percentage of citizens with basic secondary education.

It can then be seen that 17% of the population has an education of 16 years corresponding to a degree in technology.

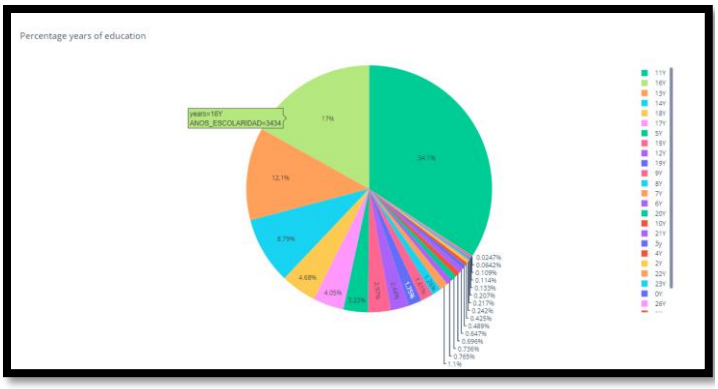


Fig137: Graph Percentage of citizens with technological education

12.1% of the population has a bachelor's degree, i.e. their level of education corresponds to secondary education.

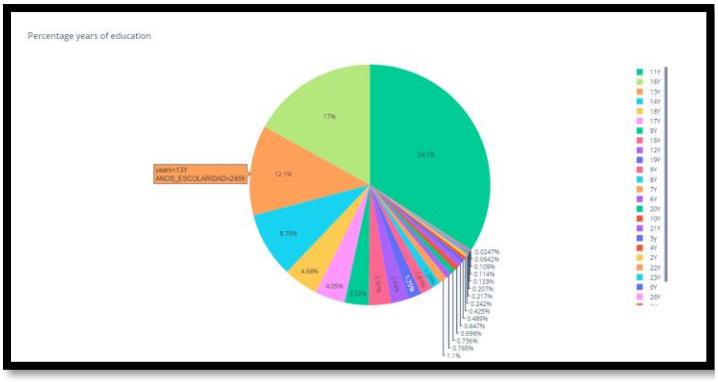


Fig138: Graph Percentage of citizens with secondary education

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

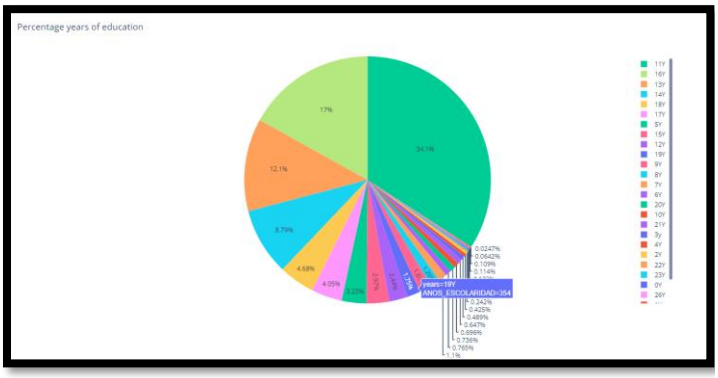


Fig139: Graph Percentage of citizens with professional training

It also shows that less than 0.1% of the population has postgraduate studies. Finally, only 0.332% of the population has doctoral studies.

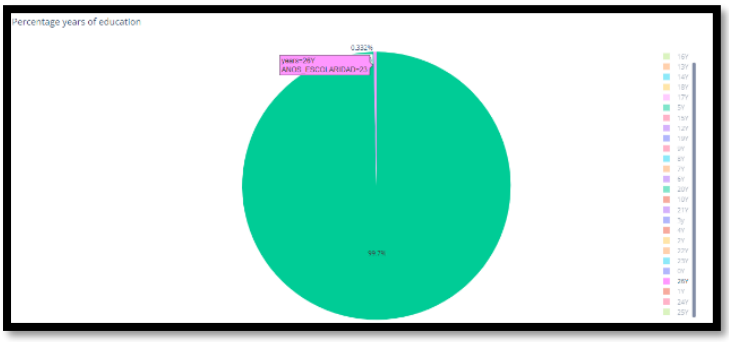


Fig140: Graph Percentage of citizens with doctoral education

Different analyses were implemented through the application of summary and agg functions.

Relationship between years of schooling and (min) housing stratum:

The relationship between years of schooling and the minimum value of the housing stratum to which the citizen belongs is presented below.

Although it makes sense that the lower the number of years of schooling, the lower the minimum value of the stratum to which one belongs, it is also noticeable that the higher the number of years of schooling, specifically in the range of (20-25) years of education, the more citizens are in strata 1 and 2.

It is even observed that citizens with 26 years of education and training belong to the lowest stratum 3.

This could have several explanations. Firstly, these values could correspond to outliers that can be analyzed in a box plot.

On the other hand, these values could be a reflection of a situation that is present in the country; "Colombia has 5.2 times more job demand than job offers. In 2019 alone, 420,000 job offers were generated and there were about 1.5 million higher education graduates looking for work" DANE (2020).

Taking into account the above, it is a worrying fact that academic training does not generate social mobility, as this closes, even more, the possibilities for the population of lower strata 1,2,3 in a state of poverty and vulnerability to improve their quality of

life, which increases the complex situation of inequality analyzed throughout this article.

```
df2 = pd.read_csv('GEIH.csv', index_col=0)
groupby_min =df2.groupby('ANOS_ESCOLARIDAD').ESTRATO_VIVIENDA.min()
print(groupby_min)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\groupby.py
ANOS_ESCOLARIDAD
0.0     1
1.0     1
2.0     1
3.0     1
4.0     1
5.0     0
6.0     1
7.0     1
8.0     1
9.0     0
10.0    1
11.0    0
12.0    1
13.0    0
14.0    0
15.0    1
16.0    1
17.0    1
18.0    1
19.0    1
20.0    1
21.0    1
22.0    2
23.0    1
24.0    2
25.0    2
26.0    3
```

Fig140: Years of schooling with respect to (min) housing stratum code.

Extending the analysis through the agg() function

The agg() function allows different functions to be executed simultaneously on a DataFrame.

The functions implemented are:

- np.len: Returns the length of a list of values.

- np.min: Returns the minimum of a list of values.

- np.max: Returns the maximum of a list of values.

- np.mean: Returns the mean of a list of values.

- np.std: Returns the standard deviation of a list of values.

We proceed to analyze the variables 'ANOS_ESCOLARIDAD' and 'N_HIJOS' with respect to the socio-economic stratum through the agg () method.

The variable N_CHILDREN corresponds to the number of children that a Colombian citizen who participated in the GEIH 2021 has.

```
df2 = pd.read_csv('GEIH.csv', index_col=0)
df2Estrato_Vivienda =df2.groupby(['ANOS_ESCOLARIDAD', 'N_HIJOS',]).ESTRATO_VIVIENDA.agg([len, max, min, np.mean, np.std])
pd.options.display.max_rows = None
print(df2Estrato_Vivienda)
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\groupby.py
                                 len  max  min      mean       std
ANOS_ESCOLARIDAD N_HIJOS
0.0              0                 8    3    1  2.250000  0.886405
                 1                 5    3    2  2.400000  0.547723
                 2                10    6    1  2.200000  1.549193
                 3                 3    2    1  1.333333  0.577350
                 4                 1    1    1  1.000000       NaN
1.0              0                 6    6    1  3.333333  2.160247
                 1                 4    3    1  2.000000  0.816497
                 2                 6    2    1  1.666667  0.516398
                 3                 3    3    1  2.000000  1.000000
                 4                 3    2    1  1.333333  0.577350
2.0              0                13    5    1  2.230769  1.165751
                 1                12    3    1  1.750000  0.621582
                 2                17    5    1  1.882353  0.992620
                 3                 6    3    1  1.500000  0.836660
                 4                 1    1    1  1.000000       NaN
3.0              0                24    5    1  2.083333  1.138904
                 1                28    5    1  2.000000  1.122167
                 2                30    5    1  1.833333  1.085431
                 3                12    3    1  1.833333  0.717741
                 4                 3    1    1  1.000000  0.000000
                 5                 2    2    1  1.500000  0.707107
4.0              0                23    5    1  1.956522  0.928256
                 1                30    3    1  2.000000  0.694808
                 2                22    9    1  2.227273  1.688387
                 3                11    3    1  1.727273  0.646670
5.0              0               155    9    1  2.219355  1.163687
                 1               211    6    0  1.976303  0.891445
                 2               193    6    1  1.974093  0.943372
                 3                73    3    1  1.863014  0.751204
                 4                14    3    1  1.714286  0.726273
                 5                 7    4    1  2.142857  1.214986
                 6                 1    1    1  1.000000       NaN
6.0              0                46    6    1  2.326087  0.967341
                 1                41    4    1  2.097561  0.888957
                 2                43    6    1  2.093023  1.042295
                 3                17    6    1  2.352941  1.169464
                 4                 6    2    1  1.500000  0.547723
                 5                 1    1    1  1.000000       NaN
                 6                 1    1    1  1.000000       NaN
7.0              0                55    5    1  2.109091  0.761887
                 1                75    9    1  2.133333  1.166345
                 2                59    6    1  2.169492  1.019678
                 3                25    6    1  1.880000  1.092398
                 4                 6    3    1  1.833333  0.752773
                 5                 1    1    1  1.000000       NaN
                 6                 1    1    1  1.000000       NaN
                 8                 1    3    3  3.000000       NaN
```

8.0	0	45	6	1	2.355556	1.047845
	1	83	6	1	2.192771	0.875936
	2	81	4	1	1.901235	0.784298
	3	32	3	1	1.656250	0.601577
	4	10	3	1	1.900000	0.737865
	5	3	3	1	1.666667	1.154701
	6	2	2	1	1.500000	0.707107
9.0	0	75	5	1	2.240000	0.912970
	1	116	6	1	2.017241	0.884482
	2	81	5	0	1.950617	0.850018
	3	41	3	1	1.926829	0.787091
	4	8	3	1	2.000000	0.534522
	5	4	2	1	1.250000	0.500000
	6	2	2	1	1.500000	0.707107
10.0	0	34	6	1	2.058824	1.042758
	1	47	9	1	2.212766	1.267249
	2	34	6	1	2.029412	0.999554
	3	14	3	1	1.785714	0.801784
	4	9	2	1	1.555556	0.527046
	5	3	1	1	1.000000	0.000000
11.0	0	1381	6	0	2.256336	0.898976
	1	2362	9	0	2.165538	0.871759
	2	2179	9	0	2.119780	0.861304
	3	788	9	1	1.954315	0.879207
	4	146	5	1	1.924658	0.752794
	5	44	6	1	1.977273	0.901901
	6	7	3	1	1.571429	0.786796
	7	2	3	1	2.000000	1.414214
	8	2	3	3	3.000000	0.000000
	9	2	3	3	3.000000	0.000000

12.0	0	104	5	1	2.413462	0.909610
	1	179	5	1	2.402235	0.890143
	2	158	5	1	2.322785	0.897761
	3	43	5	1	2.232558	0.840556
	4	8	3	2	2.375000	0.517549
	7	1	1	1	1.000000	NaN
	8	1	1	1	1.000000	NaN
13.0	0	556	6	1	2.366906	0.858609
	1	844	9	0	2.292654	0.874520
	2	778	6	0	2.236504	0.858363
	3	233	9	1	2.137339	0.986122
	4	36	4	0	2.111111	0.854493
	5	11	3	1	1.818182	0.873863
	6	1	1	1	1.000000	NaN
14.0	0	416	6	0	2.675481	0.953358
	1	627	6	1	2.548644	0.887674
	2	544	6	1	2.525735	0.851431
	3	158	5	1	2.348101	0.859026
	4	33	3	1	2.393939	0.704423
	5	1	2	2	2.000000	NaN
15.0	0	138	6	1	2.862319	1.047709
	1	225	6	1	2.702222	0.975249
	2	175	6	1	2.628571	1.002460
	3	50	5	1	2.520000	0.838852
	4	11	4	1	2.272727	1.009050
	5	2	3	2	2.500000	0.707107

16.0	0	894	9	1	3.344519	1.131791
	1	1199	9	1	3.069224	1.174028
	2	1032	6	1	3.086240	1.136351
	3	256	6	1	2.722656	1.050319
	4	45	5	1	2.711111	0.920035
	5	5	3	1	2.200000	0.836660
	6	2	3	1	2.000000	1.414214
	8	1	3	3	3.000000	NaN
17.0	0	274	6	1	3.689781	1.126815
	1	289	6	1	3.380623	1.070649
	2	205	6	1	3.556098	1.318431
	3	46	6	1	3.304348	1.190137
	4	6	3	2	2.333333	0.516398
	7	1	3	3	3.000000	NaN
18.0	0	348	9	1	3.793103	1.170117
	1	308	6	1	3.587662	1.181609
	2	239	6	1	3.652720	1.202546
	3	43	6	2	3.651163	1.172781
	4	9	5	1	2.888889	1.269296
	7	1	3	3	3.000000	NaN
19.0	0	134	6	1	4.037313	1.197784
	1	109	6	2	3.623853	1.128659
	2	85	6	1	3.752941	1.271537
	3	25	6	1	3.160000	1.247664
	5	1	3	3	3.000000	NaN
20.0	0	58	6	2	4.017241	1.034347
	1	48	6	2	3.500000	1.202834
	2	36	6	1	3.555556	1.297127
	3	7	6	2	3.571429	1.272418

21.0	0	43	6	2	4.000000	1.112697
	1	51	6	2	3.960784	1.215908
	2	34	6	1	3.911765	1.239933
	3	3	2	1	1.666667	0.577350
22.0	0	15	6	3	4.333333	1.175139
	1	14	6	2	4.000000	0.960769
	2	10	6	3	4.200000	0.918937
	3	3	6	3	4.000000	1.732051
	4	1	3	3	3.000000	NaN
	7	1	3	3	3.000000	NaN
23.0	0	21	6	1	3.857143	1.388730
	1	13	6	3	4.153846	0.898717
	2	8	6	2	4.000000	1.603567
24.0	0	4	5	2	3.750000	1.258306
	1	6	6	2	4.000000	1.414214
	2	3	5	3	4.000000	1.000000
25.0	0	2	3	3	3.000000	0.000000
	2	2	2	2	2.000000	0.000000
	3	1	5	5	5.000000	NaN
26.0	0	10	6	3	4.200000	1.032796
	1	6	5	3	4.166667	0.752773
	2	6	6	3	4.333333	1.032796
	3	1	5	5	5.000000	NaN

Fig141: Function AGG Years of schooling and Number of children with respect to Socio-economic stratum.

In the first place, it is observed that the higher the level of schooling, the fewer the number of children in the household; this fact is more evident after nineteen (19) years of academic training, which corresponds to a professional qualification.

Likewise, the fewer the number of years of academic training, the more children there are in the household.

It is interesting to analyze the relationship between socio-economic status and the number of years of schooling. It can be seen that there is no direct relationship between years of schooling and the max value of the stratum. As can be seen throughout all output segments, there are high stratifications. Even strata 5 and 6 are present throughout the range of (0-10) years of education.

The above analysis could be interpreted in two ways. The first considers this situation as a clear indicator of how years of schooling do not always determine the level of purchasing power in Colombia.

However, based on the theory of human capital and the contributions of Mincei and Becker, this situation does not make much sense, since education in Colombia has become a fundamental instrument for social mobility.

The second interpretation is based on the problem described extensively in the theoretical framework, regarding the shortcomings of the concept of socio-economic stratification as a focalizing element of the population.

It can also be seen that the relationship between years of schooling and the min value of the stratum has a direct relationship, at least in the lower values of the variable ANOS_ESCOLARIDAD, which is an expected response: the less academic training, the lower the purchasing power.

However, as the number of years of study increases, up to the age of 19, the predominant frequency is that corresponding to stratum 1. There are even citizens classified with socio-economic stratum 1 with 10, 21, and 23 years of professional training.

It is also observed that stratum 2 is present throughout all levels of academic education, including post-graduate studies, with the exception of the range made up of citizens with post-graduate studies.

This hypothesis is justified by the low supply of formal jobs and the low quality of Colombian education. In the 2018 PISA tests, Colombia showed a reduction in performance compared to 2015 and a difference between 80 and 100 points with respect to the OECD average. This means that a 15-year-old student in the country has 2.5 fewer years of schooling than an average OECD student. More than half of students in grade 9 do not understand what they read well, and two-thirds perform at the lowest level in mathematics (Fedesarrollo, 2022).

Low academic quality directly influences the ability to generate social mobility for low-income citizens.

Once again, the lack of relevance of socio-economic stratification as a focusing element is evident.

With respect to the median we can analyze that in the range of (0 - 15) years of education, which includes technical and secondary education, the medians are between (1 - 2) with respect to the socio-economic stratum.

Only up to 16 years of education, the median value of stratum 3, corresponding to the low middle class, is present.

Likewise, up to 19 years of education, i.e. professional qualification, there is a value of stratum 4 corresponding to the middle class in the median, however, the predominant value in this classification corresponds to stratum 3.

It can also be seen that from the age of 20 years of education onwards, the median takes on the corresponding values of 3 and 4.

Only up to the values of 25 and 26 years of education does a median value of 5 appear, although it does not correspond to the predominant frequency in either of the two values.

There are two outliers with respect to the median in the population with postgraduate education. The first is at 21 years of academic training, with a median value of 1.66, and the second is at 25 years of schooling, with a median of 2.0.

Below is the graph corresponding to the analysis implemented through the agg function:

```
df2 = pd.read_csv('GEIH.csv', index_col=0)
df2Estrato_Vivienda =df2.groupby(['ANOS_ESCOLARIDAD', 'N_HIJOS',]).ESTRATO_VIVIENDA.agg([len, max, min, np.mean, np.std])
pd.options.display.max_rows = None
#print(df2Estrato_Vivienda)


df2 = pd.read_csv('GEIH.csv' , dtype='unicode', index_col= 0)
df2.ESTRATO_VIVIENDA =df2.ESTRATO_VIVIENDA.replace({"0":"Estrato_0", "1":"Estrato_1", "2":"Estrato_2",
                                                    "3":"Estrato_3", "4":"Estrato_4", "5":"Estrato_5",
                                                    "6":"Estrato_6", "9":"Ns/Nr" })

print(df2.ESTRATO_VIVIENDA)
c1 = px.histogram(df2.ESTRATO_VIVIENDA, x="min",  color = "ESTRATO_VIVIENDA", hover_name = "ANOS_ESCOLARIDAD",
                  title = ' Relationship between socio-economic stratum and cities',
                  color_discrete_map = {'Estrato_0': '#FFB6C1',
                                        'Estrato_1': '#B69OFF',
                                        'Estrato_2': '#8A2BE2',
                                        'Estrato_3': '#FF00FF',
                                        'Estrato_4': '#00FFFF',
                                        'Estrato_5': '#FF1493',
                                        'Estrato_6': '#30BFDD',
                                        'Ns/Nr': '#1E90FF'

                                        })
c1.show()
```

Fig142. Mean by socio-economic stratum together with the minimum and maximum code values.

It can be seen in the following graph that there is no mean for a min value of stratum 4 in the set analyzed, which is in agreement with what is shown in figure 141, where there is no min value of stratum 4.

It can also be seen that the min value for stratum 2 is the one with the highest number of means. This is also evident in figure 141 where the predominant stratum value for the min category is 2.

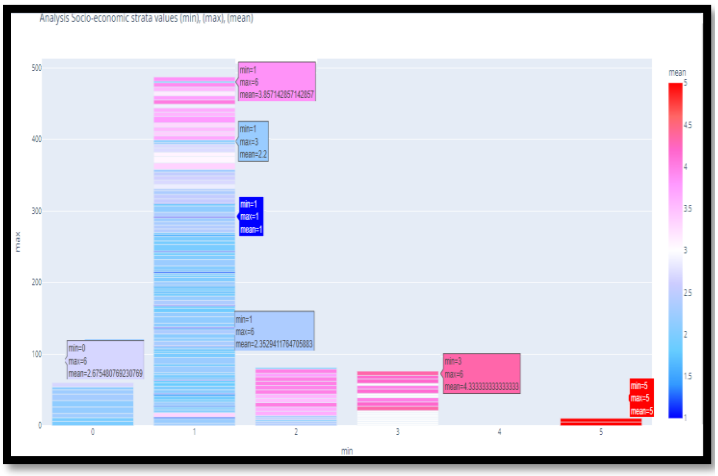


Fig 143 Mean by socio-economic stratum together with the minimum and maximum values.

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.

There are isolated cases in the values 3,10,11 and 25 years of schooling where there are records with standard deviations of 0.

Below is the graph corresponding to the behavior of the standard deviation in the Dataframe analyzed.

```
df2 = pd.read_csv('GEIH.csv' , dtype='unicode', index_col= 0)
df2.ESTRATO_VIVIENDA =df2.ESTRATO_VIVIENDA.replace({"0":"Estrato_0", "1":"Estrato_1", "2":"Estrato_2",
                                                    "3":"Estrato_3", "4":"Estrato_4", "5":"Estrato_5",
                                                    "6":"Estrato_6", "9":"Ns/Nr" })

h = px.bar(df2Estrato_Vivienda, x= 'min', y = 'max', color='std',
           title = 'Analysis Socio-economic strata values (min), (max), (std)',
           color_continuous_scale = 'picnic')

h.show()
```

Fig144. Std by socio-economic stratum together with the minimum and maximum code values.

The following graph shows the behavior of the standard deviation analyzed above with respect to the mean of the DataFrame analyzed.

Fig145. Std by socio-economic stratum together with minimum and maximum values.

The graph below shows the relationship between years of schooling and socio-economic stratum.

```
df2 = pd.read_csv('GEIH.csv' , dtype='unicode', index_col= 0)
df2.ESTRATO_VIVIENDA =df2.ESTRATO_VIVIENDA.replace({"0":"Estrato_0", "1":"Estrato_1", "2":"Estrato_2",
                                                    "3":"Estrato_3", "4":"Estrato_4", "5":"Estrato_5",
                                                    "6":"Estrato_6", "9":"Ns/Nr" })

print(df2.ESTRATO_VIVIENDA)
c1 = px.histogram(df2, x="ANOS_ESCOLARIDAD", color = "ESTRATO_VIVIENDA", hover_name = "NOMBRE_DEPTO",
                  title = ' Relationship between socio-economic stratum and years of schooling',
                  color discrete map = {'Estrato_0': '#FFB6C1',
                                        'Estrato_1': '#8690FF',
                                        'Estrato_2': '#8A2BE2',
                                        'Estrato_3': '#FF00FF',
                                        'Estrato_4': '#00FFFF',
                                        'Estrato_5': '#FF1493',
                                        'Estrato_6': '#30BFDD',
                                        'Ns/Nr': '#1E90FF'

                                        })
c1.show()
```

Fig146. 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.

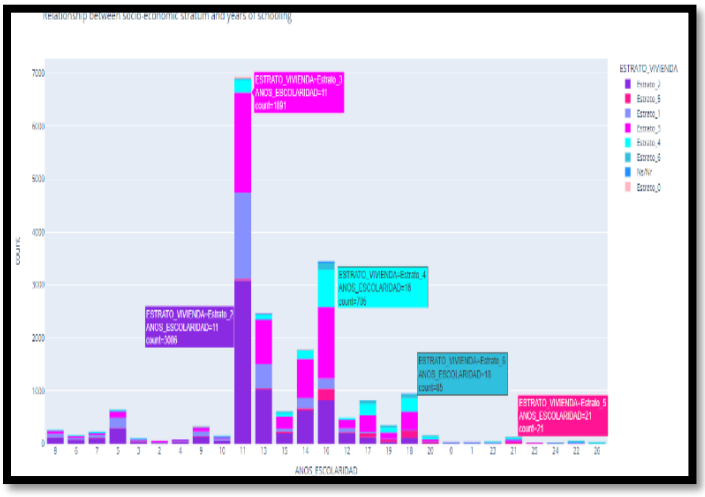


Fig147. Relationship between years of schooling and socio-economic stratum

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

```
df2 = pd.read_csv('GEIH.csv' , dtype='unicode', index_col= 0)
df2.ESTRATO_VIVIENDA =df2.ESTRATO_VIVIENDA.replace({"0":"Estrato_0", "1":"Estrato_1", "2":"Estrato_2",
                                                    "3":"Estrato_3", "4":"Estrato_4", "5":"Estrato_5",
                                                    "6":"Estrato_6", "9":"Ns/Nr" })

print(df2.ESTRATO_VIVIENDA)
c1 = px.histogram(df2, x="NOMBRE_DEPTO", color = "ESTRATO_VIVIENDA", hover_name = "ANOS_ESCOLARIDAD",
                  title = ' Relationship between socio-economic stratum and cities',
                  color_discrete_map = {'Estrato_0': '#FFB6C1',
                                        'Estrato_1': '#8690FF',
                                        'Estrato_2': '#8A2BE2',
                                        'Estrato_3': '#FF00FF',
                                        'Estrato_4': '#00FFFF',
                                        'Estrato_5': '#FF1493',
                                        'Estrato_6': '#30BFDD',
                                        'Ns/Nr': '#1E90FF'

                                        })
c1.show()
```

Fig148. Relationship between years of schooling and socio-economic stratum code

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

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.

All of the above confirms the analyses carried out previously.

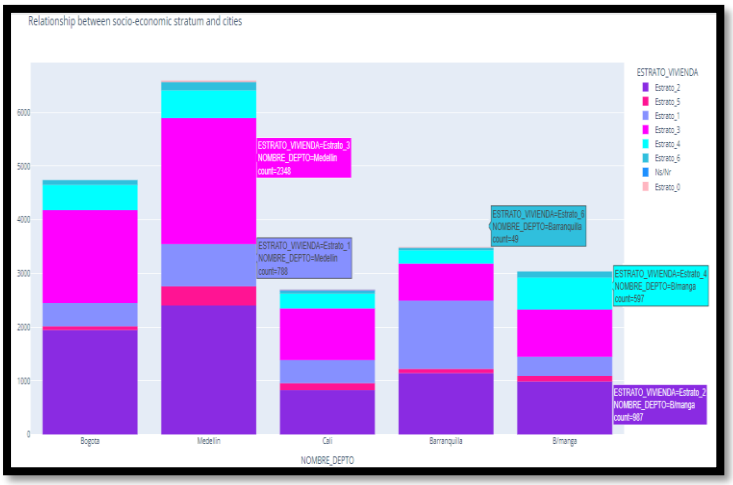


Fig149. Relationship between years of schooling and socio-economic stratum

Finally, the graph summarizing the analysis implemented in this section is generated.

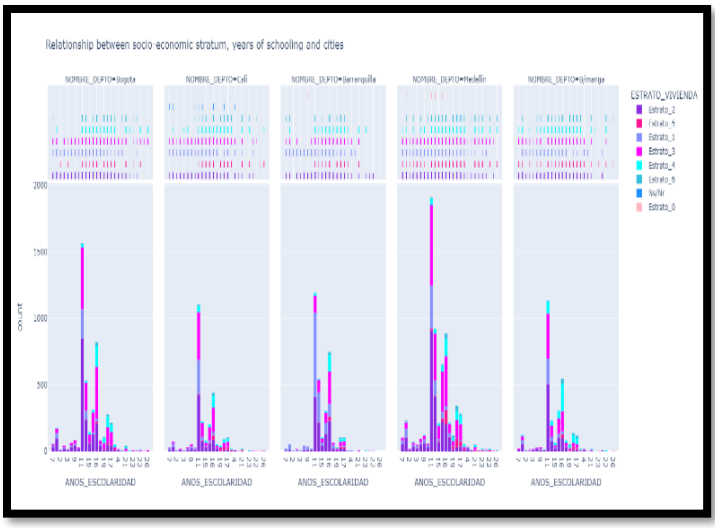


Fig150. Summary graph of implemented analysis

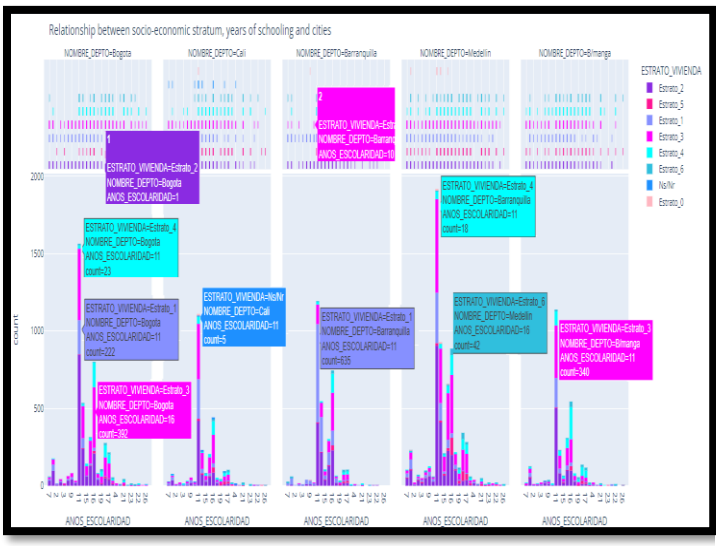


Fig151. Summary graph of the implemented analysis, and verification of values.

Next, we will study the variables working_hours_week and working_income, which are of vital importance for this analysis.

Analysis of the variable WORK_HOURS_WEEKLY

Applying the DataFrame analysis procedure we have:

```
df2 = pd.read_csv('GEIH.csv', index_col=0)
group_by_H_L_S = df2.groupby('HORAS_LABORALES_SEMANA').HORAS_LABORALES_SEMANA.count()
print(group_by_H_L_S)


sns.countplot(x =df2.HORAS_LABORALES_SEMANA, palette = sns.color_palette("pastel" ),
              saturation = 1 ). set (title = 'Working hours per week')
plt.show()
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\groupby.py
HORAS_LABORALES_SEMANA
4       2
5       1
6       4
8       6
9       1
       ..
107     1
108     1
112     2
120     2
130     2
Name: HORAS_LABORALES_SEMANA, Length: 89, dtype: int64
```

Fig152. Frequency of the column Working Hours Week code

Analyzing this variable by ranges we have:

Rank1. (0 - 36) Working hours per week

Rank2. (37 - 61) Working hours per week

Rank3. (62 - 91) Working hours per week

The respective graphs are presented below:

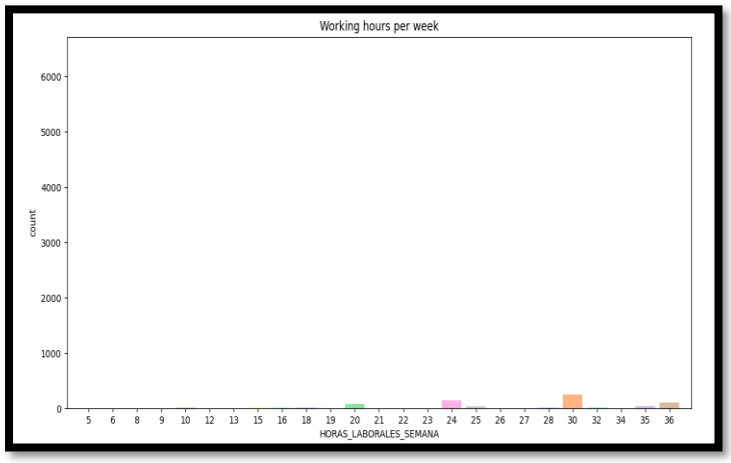


Fig.153 Range1. (0 - 36) Working hours per week.

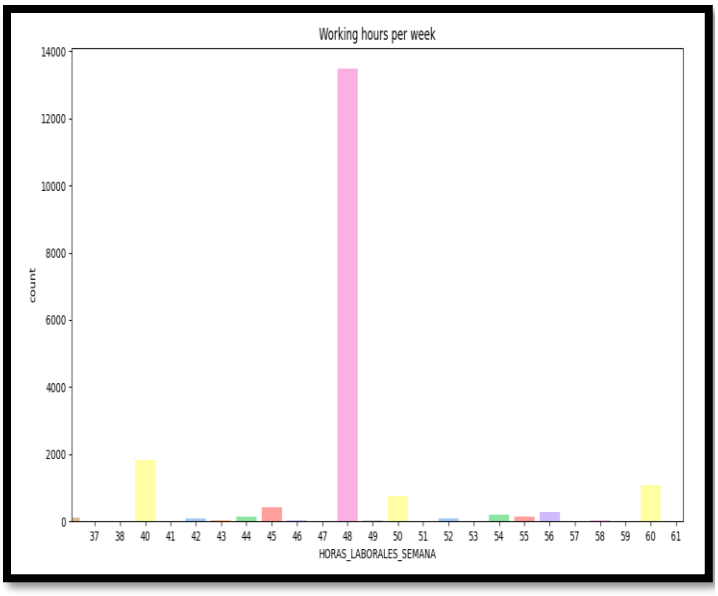


Fig154 Rank2. (37 - 61) Working hours per week.

This range 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.

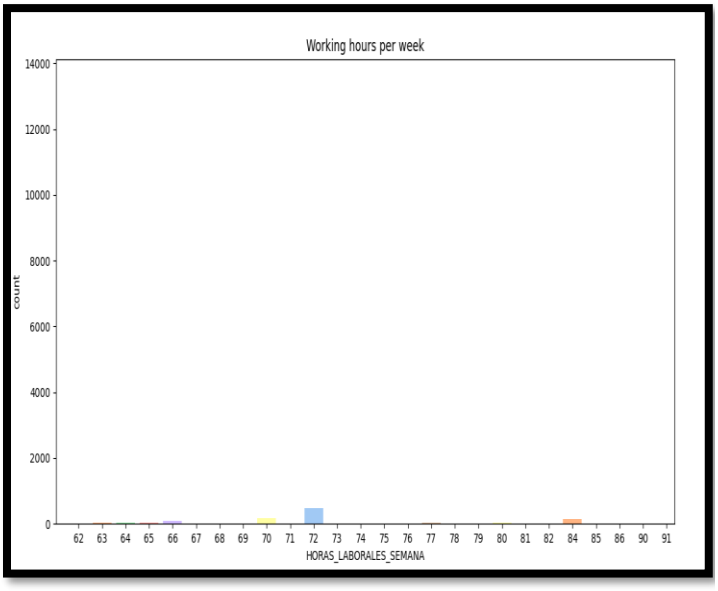


Fig155 Rank3. (62 - 91) Working hours per week.

In this range, as in range 1, there is a low distribution of data with respect to range 2; however, the highest values of frequency of occurrence are observed in 72, 84, 70, 66, 65, 64, and 63 hours.

Analysis of the variable LABOUR_INCOME

Labor income is estimated for a monthly period.

Applying the DataFrame analysis procedure we have:

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode', index_col=0)
group_by_H_L_S = df2.groupby("INGRESOS_LABORALES").INGRESOS_LABORALES.count()
print(group_by_H_L_S)
group_by_H_L_S1 =group_by_H_L_S.to_frame()
```

```
C:\Users\DELL\KAGGLE2FINALPRO>c:\Users\DELL\KAGGLE2FINALPRO\horaslLa.py
INGRESOS_LABORALES
100000        5
1000000    1057
10000000     47
1004000       3
1005000       2
           ...
9951000       1
998000        1
998350        1
998526        3
9999          1
Name: INGRESOS_LABORALES, Length: 1202, dtype: int64
```

Fig156. Frequency 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.

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode', index_col=0)
df2.ESTRATO_VIVIENDA =df2.ESTRATO_VIVIENDA.replace({"0":"Estrato_0", "1":"Estrato_1", "2":"Estrato_2",
                                                    "3":"Estrato_3", "4":"Estrato_4", "5":"Estrato_5",
                                                    "6":"Estrato_6", "9":"Ns/Nr" })


#* ojo esta grafica es vital!!!
a2 = px.histogram(df2.iloc[0:20000], x="INGRESOS_LABORALES", color = "ESTRATO_VIVIENDA",
                  hover_name = "N_PERSONAS_HOGAR", marginal = "rug", facet_col = "NOMBRE_DEPTO",
                  title = ' Relationship between socio-economic stratum, labour incomes years and cities',
                  color_discrete_map = {'Estrato_0': '#FFB6C1',
                                        'Estrato_1': '#8690FF',
                                        'Estrato_2': '#8A2BE2',
                                        'Estrato_3': '#FF00FF',
                                        'Estrato_4': '#00FFFF',
                                        'Estrato_5': '#FF1493',
                                        'Estrato_6': '#30BFDD',
                                        'Ns/Nr': '#1E90FF'

                                       })
a2.show()
```

Fig157. Relationship between the variable Labor income, socio-economic stratum, and city code.

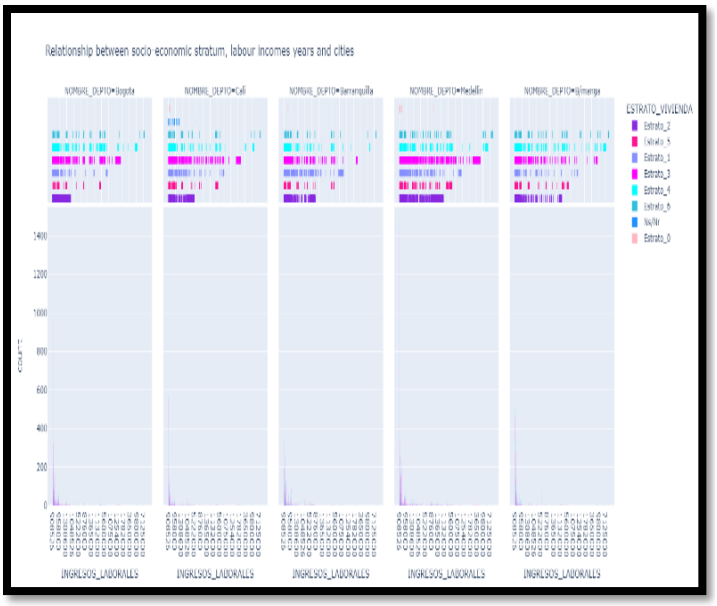


Fig158. Relationship between the variable labor income, socio-economic stratum, and city.

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.

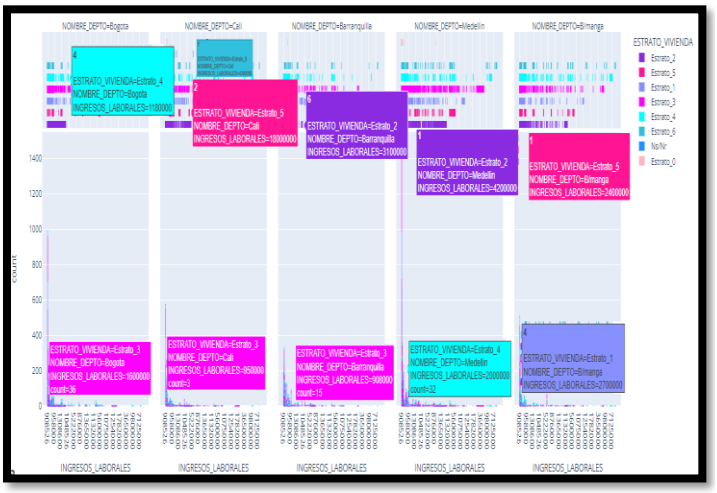


Fig159. Relationship between the variable Labor income, socio-economic stratum and city with indicator values.

It is possible to appreciate values that do not make sense even within the same city, some of the appreciations are described below.

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.

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.

Extending the analysis through the agg() function

The agg() function is applied to the DataFrame with the objective of executing different functions simultaneously.

We proceed to analyze the variables AGE_YEARS and HOUSING_STRATUM with respect to the variables LABOUR_INCOME and AVERAGE_INCOME with respect to the socio-economic stratum through the agg() method.

Average_Income corresponds to the column created within the DataFrame that divides the value of the labor income by the number of inhabitants of the

household. This gives an accurate average of the income of each household interviewed.

The following image presents the procedure.

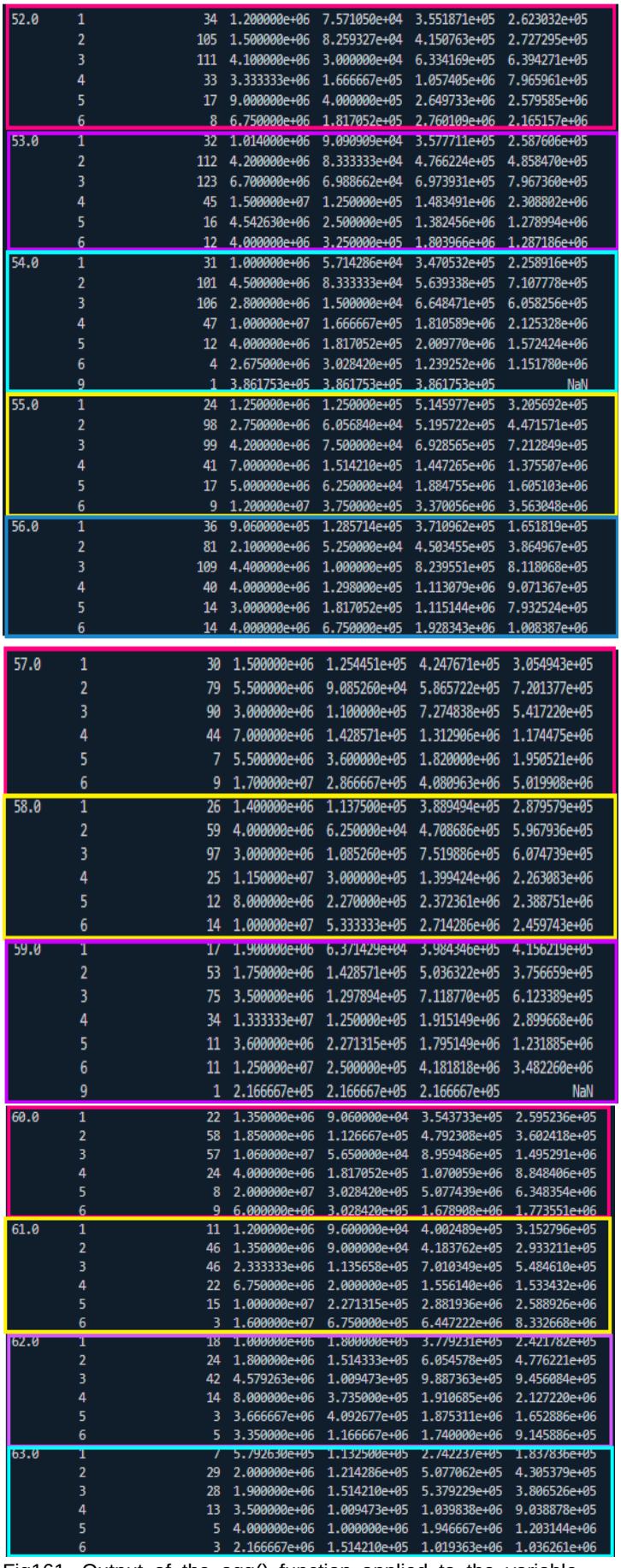
```
df2 = pd.read_csv('GEIH.csv', index_col=0)

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)
```

Fig160 agg() function applied to the variable Average_Income and number of inhabitants per household with respect to age years.

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
23.0	1	120	1.800000e+06	3.750000e+04	3.376199e+05	2.661285e+05
	2	211	2.900000e+06	1.211367e+04	3.718049e+05	3.009045e+05
	3	156	6.750000e+06	7.500000e+04	4.715461e+05	5.997251e+05
	4	35	2.000000e+06	1.166667e+05	5.706753e+05	3.666898e+05
	5	13	8.666667e+05	1.857143e+05	4.649550e+05	2.410729e+05
	6	7	1.950000e+06	1.817052e+05	8.807198e+05	6.694056e+05
24.0	1	140	2.000000e+06	3.333333e+04	3.531681e+05	2.870073e+05
	2	243	3.000000e+06	5.142857e+04	4.156523e+05	3.284498e+05
	3	190	3.800000e+06	3.250000e+04	4.700495e+05	4.242172e+05
	4	52	2.500000e+06	1.817052e+05	6.802184e+05	4.789737e+05
	5	9	1.500000e+06	3.500000e+05	8.654074e+05	4.222497e+05
	6	7	2.000000e+06	4.000000e+05	1.024762e+06	6.925499e+05
25.0	1	132	1.560000e+06	2.500000e+04	3.280776e+05	2.648204e+05
	2	305	2.500000e+06	7.571050e+04	4.175119e+05	3.235093e+05
	3	225	2.800000e+06	2.500000e+04	5.198663e+05	4.643308e+05
	4	57	3.700000e+06	1.009473e+05	8.866763e+05	7.206066e+05
	5	24	3.800000e+06	1.135750e+05	8.964950e+05	9.126517e+05
	6	9	2.500000e+06	2.000000e+05	9.882113e+05	8.796894e+05
	9	1	1.817052e+05	1.817052e+05	1.817052e+05	NaN
26.0	0	1	9.085260e+05	9.085260e+05	9.085260e+05	NaN
	1	127	4.575000e+06	6.984615e+04	3.797221e+05	4.515617e+05
	2	278	3.000000e+06	6.111111e+04	4.028679e+05	3.045232e+05
	3	228	5.400000e+06	9.085260e+04	6.475112e+05	6.314651e+05
	4	72	5.500000e+06	1.000000e+05	8.119844e+05	7.456872e+05
	5	16	9.500000e+06	2.600000e+05	1.283021e+06	2.236663e+06
	6	10	3.000000e+06	2.271315e+05	1.322365e+06	1.044418e+06
27.0	0	1	1.500000e+05	1.500000e+05	1.500000e+05	NaN
	1	125	2.250000e+06	7.566667e+04	3.511311e+05	2.768746e+05
	2	293	2.200000e+06	5.000000e+04	4.186255e+05	3.057747e+05
	3	241	3.200000e+06	8.259327e+04	5.864796e+05	4.985939e+05
	4	66	7.000000e+06	1.571429e+05	1.095252e+06	1.319834e+06
	5	19	6.000000e+06	4.542630e+05	1.939821e+06	1.791302e+06
	6	6	1.666667e+06	3.000000e+05	1.013889e+06	5.114703e+05
28.0	0	2	2.333333e+05	2.271315e+05	2.302324e+05	4.385358e+03
	1	116	4.800000e+06	5.000000e+04	3.928268e+05	4.693826e+05
	2	295	5.200000e+06	6.875000e+04	4.337915e+05	4.155450e+05
	3	223	5.000000e+06	8.125000e+04	6.216524e+05	6.326372e+05
	4	55	7.000000e+06	1.297894e+05	1.229430e+06	1.327896e+06
	5	27	5.000000e+06	2.271315e+05	1.814839e+06	1.327577e+06
	6	14	4.300000e+06	4.400000e+05	1.412924e+06	1.096778e+06
	9	3	5.625000e+05	1.920000e+05	3.515000e+05	1.905433e+05
29.0	0	1	3.028420e+05	3.028420e+05	3.028420e+05	NaN
	1	134	1.500000e+06	1.000000e+05	3.262853e+05	2.262579e+05
	2	270	4.000000e+06	7.571050e+04	4.401604e+05	4.013605e+05
	3	200	6.400000e+06	9.085260e+04	7.700780e+05	9.391853e+05
	4	77	1.100000e+07	1.000000e+05	1.436112e+06	1.766735e+06
	5	18	5.000000e+06	3.333333e+05	2.106926e+06	1.571254e+06
	6	8	7.000000e+06	4.542630e+05	2.229366e+06	2.150433e+06
	9	1	7.000000e+05	7.000000e+05	7.000000e+05	NaN
30.0	0	1	3.000000e+05	3.000000e+05	3.000000e+05	NaN
	1	136	1.330000e+06	6.489471e+04	3.335378e+05	2.334400e+05
	2	274	4.500000e+06	2.270000e+04	4.418911e+05	4.449031e+05
	3	210	4.000000e+06	3.333333e+04	6.716726e+05	6.743571e+05
	4	61	7.000000e+06	1.009473e+05	1.363553e+06	1.471734e+06
	5	18	5.500000e+06	4.500000e+05	2.030556e+06	1.312152e+06
	6	9	3.800000e+06	3.600000e+05	1.345630e+06	1.359709e+06
	9	1	2.800000e+05	2.800000e+05	2.800000e+05	NaN
31.0	0	1	3.000000e+05	3.000000e+05	3.000000e+05	NaN
	1	112	2.500000e+06	6.250000e+04	3.657257e+05	3.471511e+05
	2	270	3.500000e+06	6.000000e+04	4.719515e+05	4.742701e+05
	3	248	5.300000e+06	6.140000e+04	6.683437e+05	6.751529e+05
	4	70	8.150000e+06	1.297894e+05	1.402425e+06	1.418942e+06
	5	22	5.000000e+06	1.514210e+05	1.745687e+06	1.564033e+06
	6	15	4.600000e+06	5.500000e+05	2.462840e+06	1.433066e+06
	9	2	5.000000e+06	6.200000e+05	2.810000e+06	3.097128e+06

32.0	0	1	4.542630e+05	4.542630e+05	4.542630e+05	NaN
	1	123	1.300000e+06	5.000000e+04	3.540375e+05	2.602754e+05
	2	264	6.500000e+06	6.984615e+04	4.563058e+05	4.960616e+05
	3	209	3.500000e+06	9.085260e+04	6.639078e+05	5.749826e+05
	4	67	7.000000e+06	1.666667e+05	1.321855e+06	1.139794e+06
	5	19	4.500000e+06	3.750000e+05	2.157719e+06	1.126369e+06
	6	14	5.000000e+06	1.510000e+05	2.087214e+06	1.420701e+06
33.0	0	2	3.333333e+05	3.028420e+05	3.180877e+05	2.156063e+04
	1	103	1.280000e+06	7.200000e+03	3.410247e+05	2.404434e+05
	2	246	5.700000e+06	3.000000e+04	5.163492e+05	5.875916e+05
	3	202	5.123000e+06	1.000000e+05	7.202397e+05	7.488796e+05
	4	59	9.300000e+06	1.200000e+05	1.444991e+06	1.571697e+06
	5	20	8.500000e+06	3.000000e+05	2.129167e+06	1.999442e+06
	6	10	6.500000e+06	1.066667e+06	3.056392e+06	1.876442e+06
34.0	0	1	3.028420e+05	3.028420e+05	3.028420e+05	NaN
	1	114	3.422000e+06	6.250000e+04	3.864983e+05	3.947734e+05
	2	233	2.500000e+06	8.181818e+04	4.226583e+05	3.672036e+05
	3	211	8.100000e+06	9.681725e+04	7.946094e+05	8.797973e+05
	4	59	2.066667e+07	5.500000e+04	1.949941e+06	3.308969e+06
	5	30	9.000000e+06	3.600000e+05	2.342142e+06	2.114020e+06
	6	19	8.000000e+06	3.017052e+05	3.087809e+06	2.330155e+06
35.0	1	99	1.500000e+06	6.969231e+04	3.818562e+05	2.734015e+05
	2	204	4.000000e+06	6.489471e+04	4.826988e+05	4.410433e+05
	3	185	1.000000e+07	1.025000e+05	8.232495e+05	1.132597e+06
	4	65	1.300000e+07	1.817052e+05	1.564915e+06	1.841279e+06
	5	21	5.800000e+06	2.270000e+05	2.183200e+06	1.591681e+06
	6	13	3.900000e+06	3.160000e+05	2.077513e+06	1.118967e+06
36.0	0	1	8.750000e+04	8.750000e+04	8.750000e+04	NaN
	1	96	1.800000e+06	6.487543e+04	3.019458e+05	2.496445e+05
	2	199	3.600000e+06	7.571050e+04	4.564090e+05	4.374144e+05
	3	169	6.500000e+06	9.500000e+04	7.460425e+05	8.110352e+05
	4	73	5.300000e+06	1.500000e+05	1.049051e+06	9.741945e+05
	5	26	1.200000e+07	1.714286e+05	2.070778e+06	2.498313e+06
	6	8	2.750000e+06	1.818000e+05	1.863871e+06	1.082306e+06
37.0	1	106	1.050000e+06	8.181818e+04	2.985015e+05	1.513600e+05
	2	181	2.000000e+06	1.250000e+04	4.451857e+05	3.473115e+05
	3	187	6.000000e+06	3.333000e+03	8.211658e+05	1.002939e+06
	4	60	9.951000e+06	1.666667e+05	1.920430e+06	2.024882e+06
	5	23	1.680000e+07	1.513333e+05	3.042303e+06	3.749708e+06
	6	13	7.000000e+06	2.617052e+05	1.599599e+06	1.973198e+06
38.0	1	91	1.100000e+06	5.750000e+04	3.018927e+05	1.794274e+05
	2	215	2.800000e+06	9.085260e+04	4.322534e+05	4.098576e+05
	3	223	3.900000e+06	5.217391e+04	6.640813e+05	6.049050e+05
	4	61	1.200000e+07	2.200000e+05	1.635045e+06	2.186096e+06
	5	19	6.800000e+06	2.270000e+05	1.804202e+06	1.781555e+06
	6	9	1.600000e+07	6.000000e+05	3.249656e+06	4.911523e+06
	9	2	2.300000e+06	3.028420e+05	1.301421e+06	1.412204e+06
39.0	1	90	1.000000e+06	7.500000e+04	3.040215e+05	1.747348e+05
	2	196	9.080000e+06	6.500000e+04	4.901788e+05	7.688006e+05
	3	190	3.777000e+06	7.500000e+04	6.777342e+05	6.089825e+05
	4	47	4.400000e+06	1.755606e+05	1.144914e+06	9.450018e+05
	5	25	8.000000e+06	2.271315e+05	2.326779e+06	1.802420e+06
	6	11	1.147233e+07	4.500000e+05	3.777030e+06	3.144949e+06
40.0	1	88	1.600000e+06	6.969231e+04	3.634140e+05	2.768541e+05
	2	190	2.000000e+06	5.136000e+04	4.106000e+05	3.149702e+05
	3	186	1.350000e+07	3.000000e+04	8.120184e+05	1.259399e+06
	4	60	8.333333e+06	1.000000e+05	1.843349e+06	1.983912e+06
	5	28	1.500000e+07	4.000000e+05	2.512202e+06	2.786337e+06
	6	18	1.000000e+07	3.117052e+05	2.990157e+06	2.933058e+06
41.0	1	83	1.400000e+06	1.000000e+05	3.331713e+05	2.327602e+05
	2	188	3.250000e+06	5.687500e+04	4.404785e+05	4.021773e+05
	3	160	4.000000e+06	1.297894e+05	7.647911e+05	7.555164e+05
	4	56	7.500000e+06	1.633333e+05	1.450037e+06	1.549420e+06
	5	23	6.250000e+06	2.270640e+05	1.624352e+06	1.728190e+06
	6	8	7.500000e+06	3.000000e+05	2.400000e+06	2.452950e+06
42.0	1	67	1.500000e+06	1.000000e+05	3.521161e+05	2.357201e+05
	2	174	2.725578e+06	8.461538e+04	4.286761e+05	3.239785e+05
	3	153	7.000000e+06	5.000000e+04	6.963693e+05	9.292910e+05
	4	61	4.800000e+06	1.400000e+05	1.197633e+06	1.116504e+06
	5	14	5.750000e+06	3.785525e+05	1.993706e+06	1.605408e+06
	6	12	8.000000e+06	1.135658e+05	2.224047e+06	2.294547e+06
43.0	1	44	1.000000e+06	9.085260e+04	2.900358e+05	2.136496e+05
	2	156	4.000000e+06	2.000000e+04	4.773262e+05	5.572885e+05
	3	146	6.000000e+06	2.500000e+04	8.413173e+05	1.038959e+06
	4	45	2.000000e+07	1.333333e+05	2.202268e+06	3.448454e+06
	5	25	9.085260e+06	2.833333e+05	2.245156e+06	2.192945e+06
	6	18	1.166667e+07	2.416667e+05	2.936924e+06	3.021285e+06
44.0	1	78	1.200000e+06	1.009473e+05	3.023404e+05	1.811666e+05
	2	138	2.836000e+06	5.416667e+04	4.619773e+05	4.638759e+05
	3	137	5.700000e+06	1.011111e+05	7.351102e+05	8.263144e+05
	4	47	4.000000e+06	1.375000e+05	1.208034e+06	9.664131e+05
	5	19	3.500000e+06	2.816000e+05	1.351109e+06	9.168099e+05
	6	13	1.040000e+07	3.000000e+05	2.907764e+06	2.755674e+06
45.0	1	66	1.450000e+06	6.714286e+04	3.058493e+05	2.021071e+05
	2	134	3.000000e+06	9.800000e+04	4.572965e+05	4.221206e+05
	3	122	3.000000e+06	1.135658e+05	6.729661e+05	6.024261e+05
	4	43	5.000000e+06	1.755606e+05	1.063413e+06	1.157636e+06
	5	15	7.000000e+06	2.321315e+05	2.017631e+06	1.762418e+06
	6	11	1.500000e+07	1.817052e+05	3.251410e+06	4.654220e+06
46.0	1	50	1.800000e+06	1.312500e+05	3.526366e+05	2.893352e+05
	2	121	4.666667e+06	3.000000e+04	4.596617e+05	5.256728e+05
	3	119	3.000000e+06	9.333333e+04	5.741840e+05	4.710634e+05
	4	38	6.200000e+06	2.333333e+05	1.585206e+06	1.283193e+06
	5	9	3.333333e+06	7.333333e+05	1.683333e+06	8.396014e+05
	6	14	2.250000e+07	3.333333e+05	4.080952e+06	5.965131e+06
	9	1	1.135658e+05	1.135658e+05	1.135658e+05	NaN
47.0	1	58	2.800000e+06	3.750000e+04	3.684258e+05	3.798945e+05
	2	124	1.433333e+06	1.135658e+05	3.886062e+05	2.513668e+05
	3	117	6.000000e+06	5.714286e+04	7.198083e+05	9.090094e+05
	4	46	8.311500e+06	2.271315e+05	1.869973e+06	1.962884e+06
	5	17	5.500000e+06	5.000000e+05	2.090196e+06	1.454531e+06
	6	13	5.000000e+06	3.028420e+05	1.928296e+06	1.486769e+06
48.0	1	47	7.100000e+05	7.200000e+03	2.860595e+05	1.517960e+05
	2	110	3.166667e+06	4.166667e+04	4.521149e+05	4.033695e+05
	3	124	5.000000e+06	1.000000e+05	7.180340e+05	7.564810e+05
	4	51	6.800000e+06	1.714286e+05	1.327140e+06	1.310745e+06
	5	13	1.000000e+07	5.000000e+05	2.891359e+06	2.876380e+06
	6	11	1.250000e+07	5.000000e+05	2.451145e+06	3.428510e+06
49.0	1	40	9.600000e+05	9.090909e+04	3.678184e+05	2.078860e+05
	2	96	2.000000e+06	8.909091e+04	4.389094e+05	3.583528e+05
	3	121	1.000000e+07	7.571050e+04	8.596295e+05	1.129460e+06
	4	40	9.300000e+06	1.463000e+05	1.619237e+06	1.887405e+06
	5	15	2.500000e+06	1.816000e+05	9.701382e+05	6.659132e+05
	6	9	7.000000e+07	2.500000e+05	1.062685e+07	2.242390e+07
50.0	1	36	1.666667e+06	7.550000e+04	4.378489e+05	3.402011e+05
	2	113	1.800000e+06	8.259327e+04	4.620805e+05	3.526515e+05
	3	117	4.000000e+06	1.111111e+05	6.779360e+05	7.066621e+05
	4	36	2.200000e+07	2.500000e+04	2.206270e+06	4.067092e+06
	5	16	1.400000e+07	3.000000e+05	2.487633e+06	3.320661e+06
	6	11	7.000000e+06	2.855606e+05	1.781233e+06	2.155835e+06
51.0	1	46	2.000000e+06	6.984615e+04	3.617545e+05	3.206526e+05
	2	115	3.100000e+06	3.636364e+04	4.591750e+05	4.466037e+05
	3	119	4.800000e+06	9.218182e+04	7.484405e+05	7.351091e+05
	4	35	7.000000e+06	1.816000e+05	1.264539e+06	1.298174e+06
	5	12	6.000000e+06	3.300000e+05	1.866806e+06	1.685117e+06
	6	12	5.800000e+06	4.000000e+05	2.605413e+06	1.529002e+06
	9	1	3.450000e+05	3.450000e+05	3.450000e+05	NaN



52.0	1	34	1.200000e+06	7.571050e+04	3.551871e+05	2.623032e+05
	2	105	1.500000e+06	8.259327e+04	4.150763e+05	2.727295e+05
	3	111	4.100000e+06	3.000000e+04	6.334169e+05	6.394271e+05
	4	33	3.333333e+06	1.666667e+05	1.057405e+06	7.965961e+05
	5	17	9.000000e+06	4.000000e+05	2.649733e+06	2.579585e+06
	6	8	6.750000e+06	1.817052e+05	2.760109e+06	2.165157e+06
53.0	1	32	1.014000e+06	9.090909e+04	3.577711e+05	2.587606e+05
	2	112	4.200000e+06	8.333333e+04	4.766224e+05	4.858470e+05
	3	123	6.700000e+06	6.988662e+04	6.973931e+05	7.967360e+05
	4	45	1.500000e+07	1.250000e+05	1.483491e+06	2.308802e+06
	5	16	4.542630e+06	2.500000e+05	1.382456e+06	1.278994e+06
	6	12	4.000000e+06	3.250000e+05	1.803966e+06	1.287186e+06
54.0	1	31	1.000000e+06	5.714286e+04	3.470532e+05	2.258916e+05
	2	101	4.500000e+06	8.333333e+04	5.639338e+05	7.107778e+05
	3	106	2.800000e+06	1.500000e+04	6.648471e+05	6.058256e+05
	4	47	1.000000e+07	1.666667e+05	1.810589e+06	2.125328e+06
	5	12	4.000000e+06	1.817052e+05	2.009770e+06	1.572424e+06
	6	4	2.675000e+06	3.028420e+05	1.239252e+06	1.151780e+06
	9	1	3.861753e+05	3.861753e+05	3.861753e+05	NaN
55.0	1	24	1.250000e+06	1.250000e+05	5.145977e+05	3.205692e+05
	2	98	2.750000e+06	6.056840e+04	5.195722e+05	4.471571e+05
	3	99	4.200000e+06	7.500000e+04	6.928565e+05	7.212849e+05
	4	41	7.000000e+06	1.514210e+05	1.447265e+06	1.375507e+06
	5	17	5.000000e+06	6.250000e+04	1.884755e+06	1.605103e+06
	6	9	1.200000e+07	3.750000e+05	3.370056e+06	3.563048e+06
56.0	1	36	9.060000e+05	1.285714e+05	3.710962e+05	1.651819e+05
	2	81	2.100000e+06	5.250000e+04	4.503455e+05	3.864967e+05
	3	109	4.400000e+06	1.000000e+05	8.239551e+05	8.118068e+05
	4	40	4.000000e+06	1.298000e+05	1.113079e+06	9.071367e+05
	5	14	3.000000e+06	1.817052e+05	1.115144e+06	7.932524e+05
	6	14	4.000000e+06	6.750000e+05	1.928343e+06	1.008387e+06
57.0	1	30	1.500000e+06	1.254451e+05	4.247671e+05	3.054943e+05
	2	79	5.500000e+06	9.085260e+04	5.865722e+05	7.201377e+05
	3	90	3.000000e+06	1.100000e+05	7.274838e+05	5.417220e+05
	4	44	7.000000e+06	1.428571e+05	1.312906e+06	1.174475e+06
	5	7	5.500000e+06	3.600000e+05	1.820000e+06	1.950521e+06
	6	9	1.700000e+07	2.866667e+05	4.080963e+06	5.019908e+06
58.0	1	26	1.400000e+06	1.137500e+05	3.889494e+05	2.879579e+05
	2	59	4.000000e+06	6.250000e+04	4.708686e+05	5.967936e+05
	3	97	3.000000e+06	1.085260e+05	7.519886e+05	6.074739e+05
	4	25	1.150000e+07	3.000000e+05	1.399424e+06	2.263083e+06
	5	12	8.000000e+06	2.270000e+05	2.372361e+06	2.388751e+06
	6	14	1.000000e+07	5.333333e+05	2.714286e+06	2.459743e+06
59.0	1	17	1.900000e+06	6.371429e+04	3.984346e+05	4.156219e+05
	2	53	1.750000e+06	1.428571e+05	5.036322e+05	3.756659e+05
	3	75	3.500000e+06	1.297894e+05	7.118770e+05	6.123389e+05
	4	34	1.333333e+07	1.250000e+05	1.915149e+06	2.899668e+06
	5	11	3.600000e+06	2.271315e+05	1.795149e+06	1.231885e+06
	6	11	1.250000e+07	2.500000e+05	4.181818e+06	3.482260e+06
	9	1	2.166667e+05	2.166667e+05	2.166667e+05	NaN
60.0	1	22	1.350000e+06	9.060000e+04	3.543733e+05	2.595236e+05
	2	58	1.850000e+06	1.126667e+05	4.792308e+05	3.602418e+05
	3	57	1.060000e+07	5.650000e+04	8.959486e+05	1.495291e+06
	4	24	4.000000e+06	1.817052e+05	1.070059e+06	8.848406e+05
	5	8	2.000000e+07	3.028420e+05	5.077439e+06	6.348354e+06
	6	9	6.000000e+06	3.028420e+05	1.678908e+06	1.773551e+06
61.0	1	11	1.200000e+06	9.600000e+04	4.002489e+05	3.152796e+05
	2	46	1.350000e+06	9.000000e+04	4.183762e+05	2.933211e+05
	3	46	2.333333e+06	1.135658e+05	7.010349e+05	5.484610e+05
	4	22	6.750000e+06	2.000000e+05	1.556140e+06	1.533432e+06
	5	15	1.000000e+07	2.271315e+05	2.881936e+06	2.588926e+06
	6	3	1.600000e+07	6.750000e+05	6.447222e+06	8.332668e+06
62.0	1	18	1.000000e+06	1.800000e+05	3.779231e+05	2.421782e+05
	2	24	1.800000e+06	1.514333e+05	6.054578e+05	4.776221e+05
	3	42	4.579263e+06	1.009473e+05	9.887363e+05	9.456084e+05
	4	14	8.000000e+06	3.735000e+05	1.910685e+06	2.127220e+06
	5	3	3.666667e+06	4.092677e+05	1.875311e+06	1.652886e+06
	6	5	3.350000e+06	1.166667e+06	1.740000e+06	9.145886e+05
63.0	1	7	5.792630e+05	1.132500e+05	2.742237e+05	1.837836e+05
	2	29	2.000000e+06	1.214286e+05	5.077062e+05	4.305379e+05
	3	28	1.900000e+06	1.514210e+05	5.379229e+05	3.806526e+05
	4	13	3.500000e+06	1.009473e+05	1.039838e+06	9.038878e+05
	5	5	4.000000e+06	1.000000e+06	1.946667e+06	1.203144e+06
	6	3	2.166667e+06	1.514210e+05	1.019363e+06	1.036261e+06

Fig161. Output of the agg() function applied to the variable Average_Income and number of inhabitants per household with respect to age years.

It is observed that the years from which results are obtained in this dataset is 18 years of age. However, in Colombia, the working age population (WAP) is made up of people aged 12 and over in urban areas and 10 and over in rural areas (DANE, 2023).

In Colombia, about 523,000 minors between 5 and 17 years of age work, according to the Food and Agriculture Organization of the United Nations (FAO).

The FAO collects data from the National Administrative Department of Statistics (DANE), according to which the child labor rate for the quarter October-December 2020 amounted to 4.9% in the South American country, which represents this figure. Of these young people, 242,000 are located in populated centers, while the remaining 281,000 are in dispersed rural areas.

"Child labor perpetuates the cycle of poverty of the children affected, their families and their communities. Without education, these children are likely to remain poor. The prevalence of child labor in agriculture violates the principles of decent work. By perpetuating poverty, it undermines efforts to achieve sustainable food security and end hunger," said Alan Bojanic, FAO Representative in Colombia (DANE, 2021).

It is also observed that the maximum age presented in the dataset is 63 years, i.e. work for older adults is not contemplated by the source of the dataset.

Work among older adults is a relevant variable since by 2020, 13.5% of the Colombian population will be over 60 years of age (DANE, 2020).

In addition to the above, the study by the Universidad Externado, entitled Participation of older adults in the market and household economies in Colombia. Presents the serious situation suffered by the elderly population in Colombia. "The research found that low personal incomes force many of the elderly to remain active in the labor force. Their jobs are predominantly informal (85%) and mostly self-employed (76%) in agricultural (29%) and commercial (25%) activities".

The above figures represent the failure of protection policies with respect to children and adolescents as well as the protection of the elderly population in the country. They are also a consequence of the lack of well-functioning targeting instruments for the poor and vulnerable population, which allow social assistance to be targeted to those who really need it.

It is relevant to know that for the year 2021, the current legal minimum wage was $ 908,526 without transport subsidy and $ 1,014,980 with transport subsidy, equivalent to approximately 225.26 USD and 251.65 USD, respectively. The above values are generated at the average dollar exchange rate for the year 2021.

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.

The report also reveals that: the average labor income of the employed population according to social classes during 2020. According to the report, people considered poor had an income of approximately $403,112 (99.95 USD) per month, while the vulnerable population received around $715,773 (177.47 USD).

On the other hand, the amount of the middle class was $1'563,274 (387.59 USD) and that of the upper class was set at approximately $6'214,118 (1540.72 USD) pesos.

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.

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.

The graph corresponding to the above analysis is presented below.

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

df2.ESTRATO_VIVIENDA =df2.ESTRATO_VIVIENDA.replace({"0":"Estrato_0", "1":"Estrato_1", "2":"Estrato_2",
                                                    "3":"Estrato_3", "4":"Estrato_4", "5":"Estrato_5",
                                                    "6":"Estrato_6", "9":"Ns/Nr" })
print(df2.ESTRATO_VIVIENDA)
df2 [ "INGRESOS_LABORALES" ]  =  df2 [ "INGRESOS_LABORALES" ] . astype ( float )
df2 [ "N_PERSONAS_HOGAR" ]  =  df2 [ "N_PERSONAS_HOGAR" ] . astype ( float )


df2['Ingreso_Promedio'] = df2['INGRESOS_LABORALES'] / df2['N_PERSONAS_HOGAR']
df2 [ "Ingreso_Promedio" ]  =  df2 [ "Ingreso_Promedio" ] . astype ( float )
print(df2.Ingreso_Promedio)

c = px.histogram(df2.iloc[0:20000], x="Ingreso_Promedio",
                 color = "ESTRATO_VIVIENDA",
                 hover_name = "ANOS_ESCOLARIDAD", marginal = "rug", facet_col = "NOMBRE_DEPTO",
                 title = 'The average income per household by strata and by city',
                    color_discrete_map = {'Estrato_0': '#FFB6C1',
                                          'Estrato_1': '#8690FF',
                                          'Estrato_2': '#8A2BE2',
                                          'Estrato_3': '#FF00FF',
                                          'Estrato_4': '#00FFFF',
                                          'Estrato_5': '#FF1493',
                                          'Estrato_6': '#30BFDD',
                                         Ns/Nr': '#1E90FF'

                                      })
c.show()
```

Fig162 Average income by stratum and city code.

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.

Fig163 Average income by stratum and city.

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.

Below are the graphs that describe the previous analysis, these were generated with the maximum and minimum values of the average salary per household with respect to the mean.

```
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)

a1 = px.histogram(df2Estrato_Vivienda1 , x ='min',  color = "mean",
                  title = ' Analysis average household income values (min), (max), (mean)')
a1.show()
```

Fig164 Average wage per household (min) and mean code.

The labels show the large difference between the mean and the average wage per household (min).

```
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)
```

```
a1 = px.histogram(df2Estrato_Vivienda1 , x ='max',  color = "mean",
             title = ' Analysis average household income values (min), (max), (mean)')
a1.show()
```

Fig165 Average wage per household (max) and mean code.

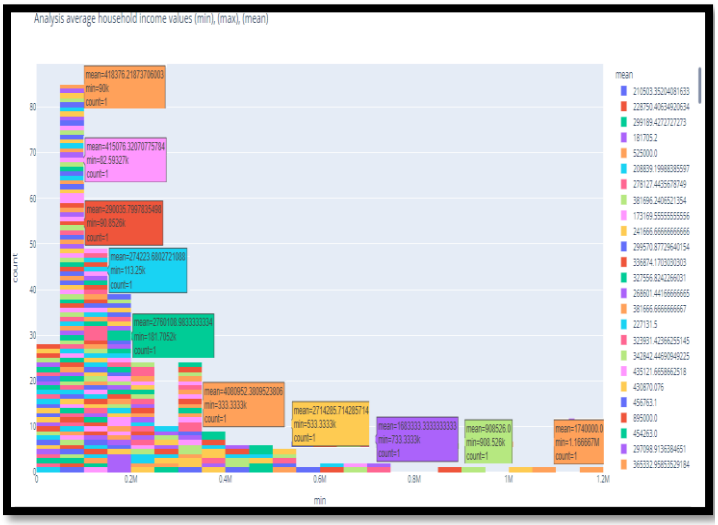


Fig166 Average wage per household (max) and mean code.

The graphs that describe the previous analysis are presented below, these were generated with the maximum and minimum values of the average wage per household with respect to the mean.

```
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)
```

```
a1 = px.histogram(df2Estrato_Vivienda1 , x ='max',  color = "mean",
             title = ' Analysis average household income values (min), (max), (mean)')
a1.show()
```

Fig167 Average wage per household (max) and mean code.

The labels show the large difference between the mean and the Average wage per household (max).

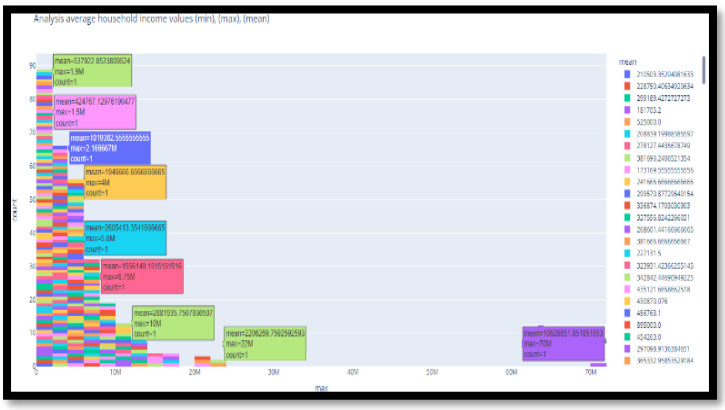


Fig168 Average wage per household (max) and mean code.

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.

```
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)
```

```
a1 = px.histogram(df2Estrato_Vivienda1 , x ='min',  color = "std",
             title = ' Analysis average household income values (min), (max), (mean)')
a1.show()
```

Fig169 Average wage per household (min) and std code.

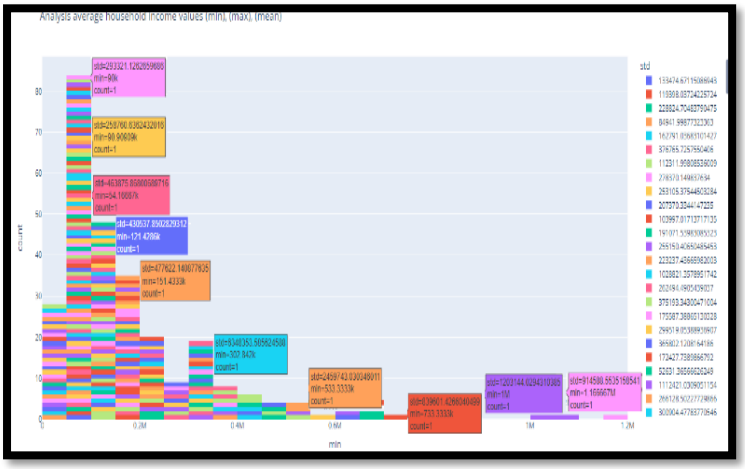


Fig170 Average wage per household (min) and std code.

```
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)

a1 = px.histogram(df2Estrato_Vivienda1 , x ='max',  color = "std",
          title = ' Analysis average household income values (min), (max), (mean)')
a1.show()
```

Fig171 Average wage per household (max) and std code.

The labels show the large dispersion of the data with respect to the mean.

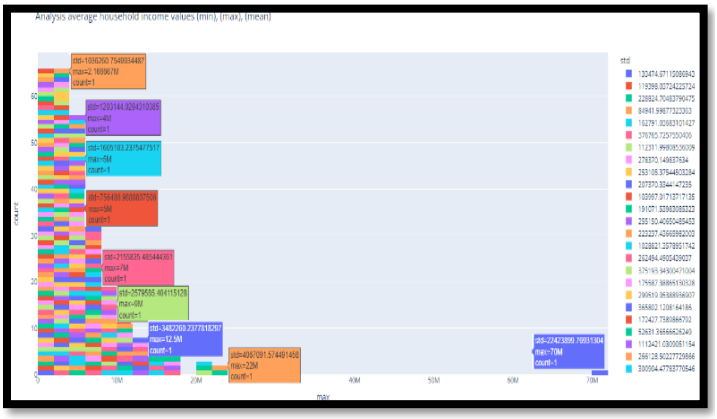


Fig172 Average wage per household (max) and std.

The above results have several explanations that are worth analyzing.

In the year 2021, the Report of the Commission of Experts on Tax Benefits for Colombia was produced. This commission stated that "the richest 1 percent of the country's wealthiest pay less income tax than one-fifth of taxpayers".

For the first time in Colombia's history, the Commission of Experts on Tax Benefits analyzed the critical situation of inequality in the country from a tax point of view.

For the commission, made up of a group of several international experts and government officials, "a person like Luis Carlos Sarmiento Angulo, the richest man in the country, pays less income tax than a taxpayer with an income in the upper middle bracket".

The main reason why the top 1% of Colombia's population in the upper social class pay less tax than even the lower classes has to do with the type of income they generate, which is different and therefore has different tax rates.

The more income a taxpayer generates, the less it depends on a salary, depending on the term (non-labor income), such as fees for professional work and those derived from private business.

These unearned incomes allow for questionable deductions "For example, there are people who report as personal expenses that have to do with the education of their children, or the housemaid, or the nanny," Luis Carlos Reyes, director of the fiscal observatory of the Javeriana University, explained to La Silla, a Colombian news outlet.

Based on the article: "This is what the richest 1% in Colombia contribute in taxes" (Forbes 2022): according to data from the Organization for Economic Co-operation and Development (OECD) with calculations from the Ministry of Finance, while a person with an average monthly income of $6 and $8 million in Colombia has an effective income tax rate of 2% and 5%, respectively, in Latin America the average is 17% and 19% for each case.

For those with an average monthly income of $42 million, the rate in Colombia is 17%, while the average in Latin America is 27%, in the United States it is 35% and in Spain, it rises to 41%. This is one of the reasons why it is requested review the rates of personal income tax paid by salaried employees in the country, as these are lower than the average for the region.

Looking at other data, Colombia is the OECD country that obtains the least resources through personal income tax. For example, Colombia collects 1.2% of GDP (2019) in this tax, while peers in the region such as Chile and Mexico collect 1.4% and 3.4% of GDP each.

An alternative tax proposal developed by the Javeriana, Externado, Rosario, Los Andes, and Friedrich-Ebert-Stiftung universities in Colombia says that $11 billion could be collected from the highest income groups in the country. For Martín Jaramillo, business consultant and university professor of economics, among the tools that the country could use to make people with higher incomes pay more taxes are the high marginal income tax rates and the tax on dividends, but the most important thing, in his words, is to modernize the Dian so that it has better data analytics and auditing capacity.

Jaramillo added that "according to the presentations of the tax expert commission, it seems that the way they avoid taxes is through legal persons. This cannot be solved by changing rates. The expert also emphasized that "the tax system, in general, is not very progressive, but a large part of this problem is due to spending: in Colombia, a lot of money is spent on subsidizing high pensions, higher education in the upper-middle deciles, and housing and public services are subsidized by strata.

As a complement to what has been said so far, Liliana Heredia, professor in the Department of Accounting and Finance at the Javeriana University in Cali and an expert on tax issues, said that perhaps the most important thing is to start pursuing tax evaders and those who hide their resources in tax havens. In other words, "it is essential to strengthening the capacity of the tax administration to carry out effective auditing". "We complain about the corruption we see in the public sector, but there is little questioning of the unethical behavior of many citizens or businessmen, and this would be a first step: to understand that the fair payment of taxes is part of our social responsibility and solidarity".

Studies by Javier Ávila and Juliana Londoño also show that a 'rich Colombian is three times more likely to evade taxes than a rich person in Scandinavia, and this should change, it should be reproached, but today it is not.

It is important to analyze the relationship between average income and years of schooling. Below is the respective graph with respect to cities.

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

df2.ESTRATO_VIVIENDA =df2.ESTRATO_VIVIENDA.replace({"0":"Estrato_0", "1":"Estrato_1", "2":"Estrato_2",
                                                    "3":"Estrato_3", "4":"Estrato_4", "5":"Estrato_5",
                                                    "6":"Estrato_6", "9":"Ns/Nr" })
print(df2.ESTRATO_VIVIENDA)
df2 [ "INGRESOS_LABORALES" ]  =  df2 [ "INGRESOS_LABORALES" ] . astype ( float )
df2 [ "N_PERSONAS_HOGAR" ]  =  df2 [ "N_PERSONAS_HOGAR" ] . astype ( float )


df2['Ingreso_Promedio'] = df2['INGRESOS_LABORALES'] / df2['N_PERSONAS_HOGAR']
df2 [ "Ingreso_Promedio" ]  =  df2 [ "Ingreso_Promedio" ] . astype ( float )
print(df2.Ingreso_Promedio)

c = px.histogram(df2.iloc[0:20000], x="Ingreso_Promedio", y = 'ANOS_ESCOLARIDAD',
           color = "ESTRATO_VIVIENDA",
           hover_name = "N_HIJOS", marginal = "rug", facet_col = "NOMBRE_DEPTO",
           title = 'The average income per household by strata and by city',
              color_discrete_map = {'Estrato_0': '#FFB6C1',
                                    'Estrato_1': '#8690FF',
                                    'Estrato_2': '#8A2BE2',
                                    'Estrato_3': '#FF00FF',
                                    'Estrato_4': '#00FFFF',
                                    'Estrato_5': '#FF1493',
                                    'Estrato_6': '#30BFDD',
                                   'Ns/Nr': '#1E90FF'

                                    })
c.show()
```

Fig173. Relationship between average income and years of schooling by city code.

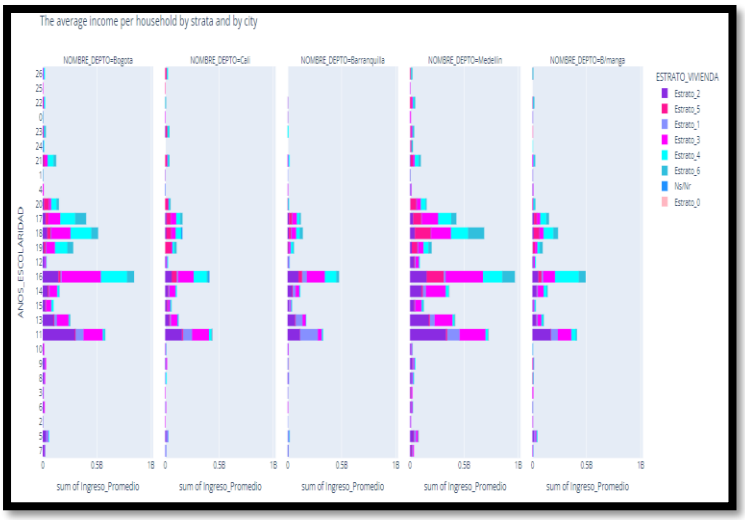


Fig174 Relationship between average income and years of schooling by city.

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.

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.

```
df2.ESTRATO_VIVIENDA =df2.ESTRATO_VIVIENDA.replace({"0":"Estrato_0", "1":"Estrato_1", "2":"Estrato_2",
                                                    "3":"Estrato_3", "4":"Estrato_4", "5":"Estrato_5",
                                                    "6":"Estrato_6", "9":"Ns/Nr" })
print(df2.ESTRATO_VIVIENDA)
df2 [ "INGRESOS_LABORALES" ]  =  df2 [ "INGRESOS_LABORALES" ] . astype ( float )
df2 [ "N_PERSONAS_HOGAR" ]  =  df2 [ "N_PERSONAS_HOGAR" ] . astype ( float )


df2['Ingreso_Promedio'] = df2['INGRESOS_LABORALES'] / df2['N_PERSONAS_HOGAR']
df2 [ "Ingreso_Promedio" ]  =  df2 [ "Ingreso_Promedio" ] . astype ( float )
print(df2.Ingreso_Promedio)

a2 = px.sunburst(df2.iloc[0:20537, ], color = "ESTRATO_VIVIENDA", values = "Ingreso_Promedio",
                 path =["NOMBRE_DEPTO", "HORAS_LABORALES_SEMANA"], hover_name = "ESTRATO_VIVIENDA",
                 title = 'Hours of work with respect to the city',
                 color_discrete_map = {'Estrato_0': '#FFB6C1',
                                       'Estrato_1': '#8690FF',
                                       'Estrato_2': '#8A2BE2',
                                       'Estrato_3': '#FF00FF',
                                       'Estrato_4': '#00FFFF',
                                       'Estrato_5': '#FF1493',
                                       'Estrato_6': '#30BFDD',
                                       'Ns/Nr': '#1E90FF'

                                       }
                 )
a2.show()
```

Fig 175 Relationship between hours worked per week and cities.

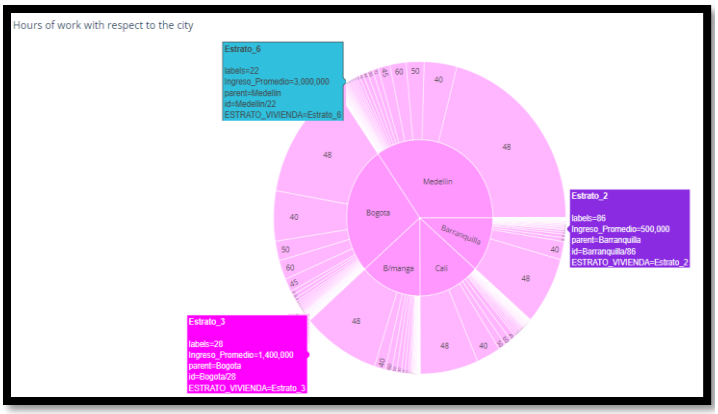


Fig176. Relation between hours worked per week and cities

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

```
df2 = pd.read_csv('GEIH.csv', dtype='unicode', index_col=0)
df2.ESTRATO_VIVIENDA =df2.ESTRATO_VIVIENDA.replace({"0":"Estrato_0", "1":"Estrato_1", "2":"Estrato_2",
                                                    "3":"Estrato_3", "4":"Estrato_4", "5":"Estrato_5",
                                                    "6":"Estrato_6", "9":"Ns/Nr" })
print(df2.ESTRATO_VIVIENDA)
df2 [ "INGRESOS_LABORALES" ]  =  df2 [ "INGRESOS_LABORALES" ] . astype ( float )
df2 [ "N_PERSONAS_HOGAR" ]  =  df2 [ "N_PERSONAS_HOGAR" ] . astype ( float )


df2['Ingreso_Promedio'] = df2['INGRESOS_LABORALES'] / df2['N_PERSONAS_HOGAR']
df2 [ "Ingreso_Promedio" ]  =  df2 [ "Ingreso_Promedio" ] . astype ( float )
print(df2.Ingreso_Promedio)

a3 = px.scatter(df2.iloc[0:20000, ], color = "ESTRATO_VIVIENDA", y = "HORAS_LABORALES_SEMANA", x ="Ingreso_Promedio", hover_name="NOMBRE_DEPTO",
                title = 'Hours of work with respect to the average income',
                color_discrete_map = {'Estrato_0': '#FFB6C1',
                                      'Estrato_1': '#8690FF',
                                      'Estrato_2': '#8A2BE2',
                                      'Estrato_3': '#FF00FF',
                                      'Estrato_4': '#00FFFF',
                                      'Estrato_5': '#FF1493',
                                      'Estrato_6': '#20BFDD',
                                      'Ns/Nr': '#1E90FF'

                                     }
)
a3.show()
```

Fig177 Relationship between hours worked per week and cities.

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.

This contrasts with the figures provided by the 2020 OECD report which found that Colombia is the member country of the community with the highest number of working hours with an average of 48 hours per week.

This study also determined that the average number of working hours per week is 20 hours per week in Latin America.

It is important to highlight that this study did not take into account a second job, which is a variable to be considered in the future analysis as it is a very frequent practice for Colombians due to the low salaries, which will generate a greater burden than the number of working hours per week. Likewise, the rural population, whose average number of working hours per week is much higher than the urban population, was not taken into account.

The concentration of average earnings is much more noticeable in the (0-2) million range.

Again, the labels show that the social class ranges do not correspond to the socio-economic stratification.

Fig 178 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.

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

print(df2.ESTRATO_VIVIENDA)
df2 [ "INGRESOS_LABORALES" ]  =  df2 [ "INGRESOS_LABORALES" ] . astype ( float )
df2 [ "N_PERSONAS_HOGAR" ]  =  df2 [ "N_PERSONAS_HOGAR" ] . astype ( float )


df2['Ingreso_Promedio'] = df2['INGRESOS_LABORALES'] / df2['N_PERSONAS_HOGAR']
df2 [ "Ingreso_Promedio" ]  =  df2 [ "Ingreso_Promedio" ] . astype ( float )
print(df2.Ingreso_Promedio)

df2 [ "ESTRATO_VIVIENDA" ] = df2 [ "ESTRATO_VIVIENDA" ] . astype ( float )
df2 [ "Ingreso_Promedio" ] = df2 [ "Ingreso_Promedio" ] . astype ( float )
df2 [ "N_PERSONAS_HOGAR" ] = df2 [ "N_PERSONAS_HOGAR" ] . astype ( float )
a3 = px.scatter(df2.iloc[0:20537, ], color = "ESTRATO_VIVIENDA" , x = "Ingreso_Promedio", y = "NOMBRE_DEPTO",
                hover_name="ANOS_ESCOLARIDAD", size = "N_PERSONAS_HOGAR" , size_max = 30,
                log_x = True, title =" Average income vs cities",
                color_continuous_scale = 'jet',
                labels = dict(
                    INGRESOS_LABORALES = "Ingresos Laaborales ($ COL)",
                    NOMBRE_DEPTO = "Nombre Departamento "))
a3.show()
```

Fig179 Relationship between average income and code cities

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.

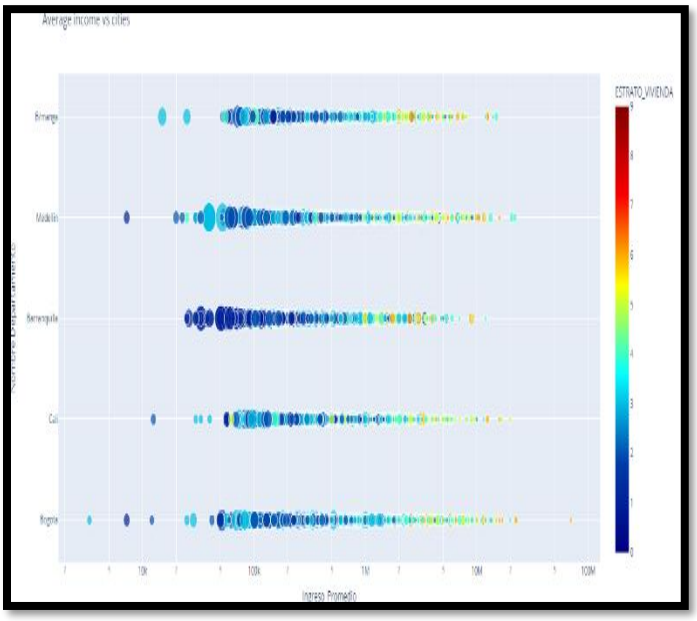


Fig180 Relationship between average income and cities

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).

This behavior is present throughout the analysis.

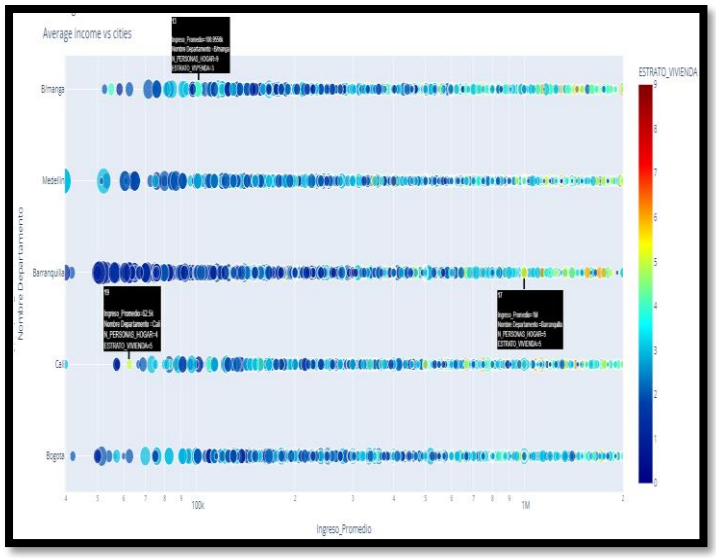


Fig181 Relationship between average income up to 2 million pesos (USD 445) and cities.

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.

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.

By being stratified in levels 1, 2, and 3, they receive subsidies and have access to social programs that have been designed by the government to support the poor and vulnerable population of the country.

Once again, the analysis of the data provided by the GEIH 2021 shows that socio-economic stratification has major shortcomings as a targeting instrument.

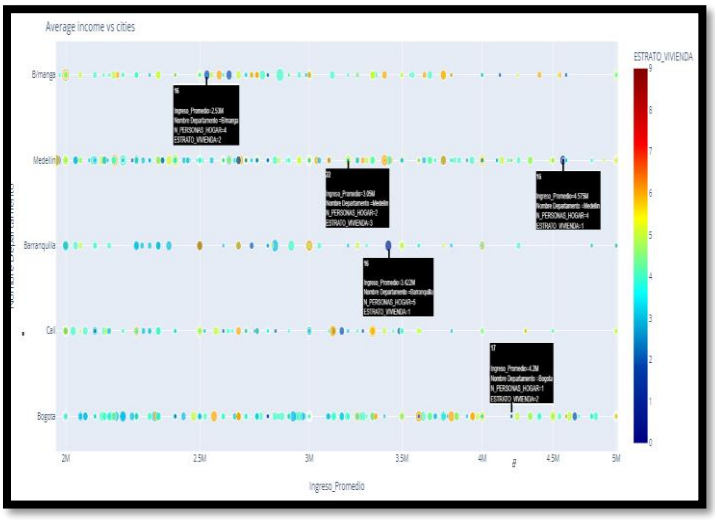


Fig 182 Ratio of average income between 2 million pesos (445.87 USD) and 5 million pesos (1239.69 USD) by city.

Inequality analysis of Colombia with regard to Latin America and the Caribbean

Next, we proceed to analyse the datasets containing the information for Colombia, Latin America, and the Caribbean.

First, we will analyze the cumulative distribution vs. income shares for the region. The following graph shows 6 categories corresponding to the classification groups given by the World Bank, broken down as follows:

Extreme Poverty: Percentage of people living on less than 2.15 USD per day (in PPP 2017).

Poverty level 1: Percentage of people living on an income range of (2.15 - 3.65) USD per day (in PPP 2017).

Poverty level 2: Percentage of people living with an income range of (3.65 - 6.85) USD, per day (in PPP 2017)

Vulnerable population: Percentage of people living with an income range of (6.85 - 14) USD per day (in PPP 2017). The vulnerable population is those who are not in poverty but have a high probability of falling into poverty in the face of any unexpected change that affects their income.

Middle class: Percentage of people living with an income range of (16 - 81) USD per day (in PPP 2017).

Rich: Percentage of people living on more than USD 81 per day (in PPP 2017).

```
df2 = pd.read_csv('DesigualdadDistribuciónDeIngresos.csv', sep=";")
df2.columns = df2.columns.str.strip()
df2['Centiles'] = pd.to_numeric(df2['Centiles'], downcast="float")
df2.columns = df2.columns.str.strip()

df2['Distribución acumulada'] = pd.to_numeric(df2['Distribución acumulada'], downcast="float")


a = px.histogram(df2, x="Distribución acumulada", y = "Proporción del ingreso", color = "País",
                 title = 'Cumulative wealth distribution vs. income shares by group',
                 hover_name = "Centiles",
                 marginal = "rug", facet_col = "Grupo" )
a.show()
```

Fig 183. Distribución acumulada vs la proporción de ingresos code.

The following graph presents the cumulative distribution vs. income spread for each of the categories established by the World Bank.

The figures are not updated with respect to the pandemic and therefore their impact cannot be analyzed in this article.

The World Bank Group aims to end extreme poverty in the world by 2030. In this regard, efforts by Latin America and the Caribbean to reduce the extreme poverty rate are noted.

The greatest amount of accumulated distribution is observed in the middle class, as well as a large percentage in the vulnerable population category. For the World Bank, an individual is defined as vulnerable if the probability of falling into poverty in the next five years is more than 10 percent.

In terms of poverty levels 1 and 2, there is a cumulative distribution with low percentages of low incomes compared to the middle class and the rich.

In the rich class, the variable cumulative distribution is observed throughout the entire range of income percentage values; this phenomenon is exclusively characteristic of this category.

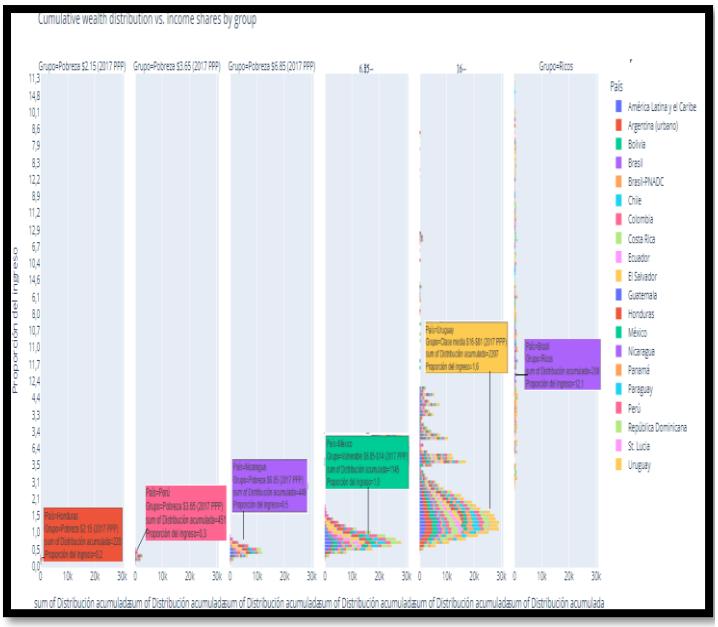


Fig 184 Cumulative distribution vs. share of income.

The following graph presents the cumulative distribution vs. income share of Colombia.

The labels allow us to analyze the abysmal differences in income shares between the classification groups proposed by the World Bank.

- Extreme Poverty: Income ratio: 0.2
- Poverty level 1: Income ratio: 0.4
- Poverty level 1: Income ratio: 0.8
- Vulnerable population: Share of income: 1.6
- Middle class: Share of income: 5.1
- Rich: Share of income: 11.7

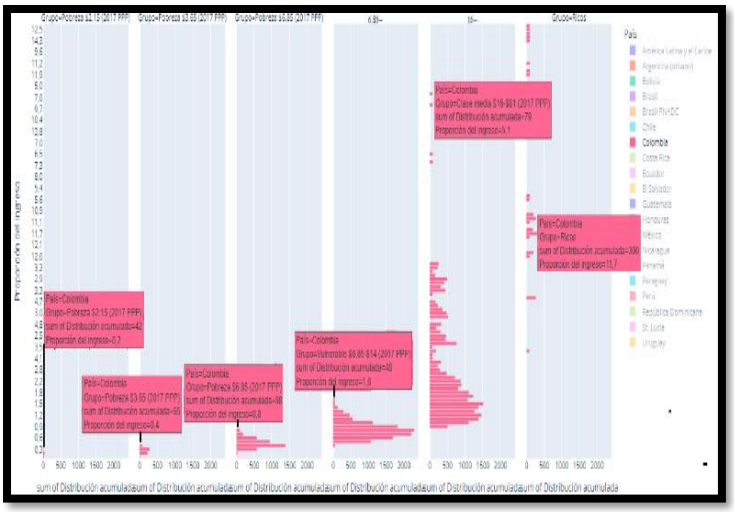


Fig185 Cumulative distribution vs. income share Colombia.

B. Predictive 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())
```

Fig 186 Creation of working DataFrame df2_Final code.

	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

Fig 187 DataFrame df2_Final code.

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

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())
```

Fig 188. Selection of the prediction target and the code features

```
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
```

Fig 189 Dataframe X.

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))
```

Fig190 Basic model implementation.

```
       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)
```

Fig 191 First prediction.

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))
```

Fig 192 Calculation of absolute error code.

```
0.4853747123436424
```

Fig 193 Absolute error.

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.

```
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))
```

Fig194. Data splitting.

```
0.8529643668585362
```

Fig195 Mean_absolute_error.

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

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

```
DecisionTreeRegressor(random_state=1)
[2.17449664 5.         2.53333333 ... 1.95652174 2.         2.86956522]
0.4853747123436424
[2.18061674 5.         2.41304348 ... 2.21428571 2.         2.9       ]
0.8529722508068175
```

Fig196 Comparison between the output of the two models

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.

The phenomenon of over-fitting tends to occur when a high value of tree depth is chosen and is characterized by a model that matches the training data almost perfectly, but performs poorly in validation and with the insertion of new data.

The phenomenon of underfitting occurs when we make the tree too shallow. As a consequence, the model does not capture important distinctions and patterns in the data, so it performs poorly even on training data.

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.

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. The implementation is presented below:

```
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)
```

```
# 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)
```

```
# 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))
```

Fig197 MAE of different values for max_leaf_nodes code.

```
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
```

Fig198. MAE of different values for max_leaf_nodes.

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).

Random Forests

Taking into account the above 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.

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.

```
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)
```

```
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))
```

Fig199. Implementation of the random forests model code.

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

Fig200 MAE of the random forests model

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.

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

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.

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.

Fiscal policy in Colombia as it is currently constituted is not a tool for reducing inequality. The low level of progressive tax collection and the multiple deductions favor the country's upper class, so there is low tax collection, which leads to minimal redistribution, which does not impact the poor and vulnerable population of the country in a forceful way because it is not properly targeted.

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

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.23.41.4105.

Bogliacino, 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.

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/National 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.