# **Tamil Nadu Marginal Workers**

# **Documentation and Submission**

**Phase :05 Project Number: 03**

**Introduction:**

As our world becomes increasingly data-driven, the importance of data analytics cannot be overstated. In this documentation, I will recount my learning experience in the "Data Analytics with Cognos" course, shedding light on the skills acquired, challenges faced, and the transformative potential of Cognos Analytics.

For analysing marginal workers in Tamil Nadu, adopt a mixed-methods approach. Combine quantitative data, such as surveys and statistical analysis, to quantify demographic patterns. Additionally, use qualitative methods like interviews or focus groups to gain deeper insights into their experiences. This hybrid approach ensures a comprehensive understanding of the socioeconomic landscape, capturing both statistical trends and personal narratives

**Course Overview:**

The "Data Analytics with Cognos" course aimed to equip students with the knowledge and skills necessary to analyze and visualize data using IBM Cognos Analytics. Taught over a 5-week (5 phase) period, the course promised to bridge the gap between theory and practice.

**Objectives:**

* IBM Cognos Analytics working (phase-1)
* Data Extraction (phase-2)
* Cognos analytics in action (phase-2)
* Python for data science (phase-3)
* Data Exploration & visualization (phase-4)
* Supervised Learning (phase-4 & 5)
* Build and deploy Machine Learning application (phase-5)

**Data Set:**

Here we are using the data set from the govt organization official site(tn.data.gov.in).

This dataset from: <https://tn.data.gov.in/resource/marginal-workers-classified-age-industrial-category-and-sex-scheduled-caste-2011-tamil>

This dataset contains 71 columns and 595 rows.

**IBM Cognos Analytics working:**

IBM Cognos Analytics is a web-based integrated business intelligence suite by IBM that provides a toolset for reporting, analytics, score carding, and monitoring of events and metrics. It is designed to help businesses make better decisions faster with AI-powered insights for everyone. Cognos Analytics enables everyone in your organization to unlock the full potential of your data with AI-powered automation and insights. The natural language AI assistant is always available, allowing you to describe the data you need and let Cognos Analytics build stunning data visualizations for you. You can also describe the question you are looking to answer or hypothesis you want to test and let AI help you get the insights you need . Cognos Analytics can help transform your business team into power users, so your data analysts can focus on finding deeper insights to boost results.

**Data Extraction:**

Data extraction is a fundamental step in the process of data analysis using IBM Cognos

* Data Source Selection:

Gain a deep understanding of the problem statement and the context of marginal workers in Tamil Nadu. Cognos can connect to a wide range of data sources, including databases, spreadsheets, and web services. You'll need to configure data connections to these sources within Cognos.

* Data Modeling:

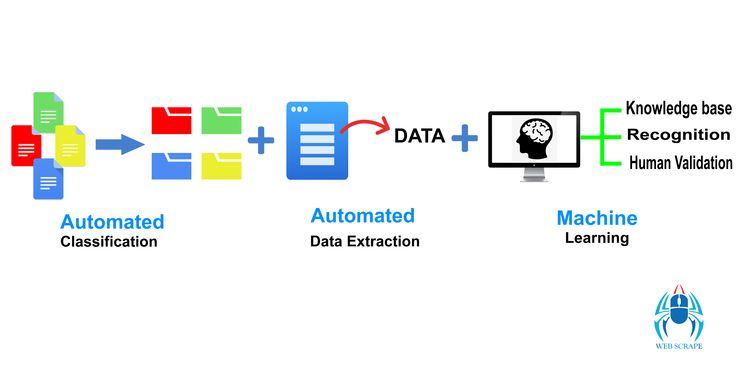
In Cognos, you can create a data model to define how the data is structured and organized. This involves identifying the tables, views, and relationships between data sources. This step helps ensure that the extracted data is in a format suitable for analysis.

* Query Building:

Once your data model is set up, you can build queries to extract the specific data you need. Cognos provides a visual query builder interface to help you design and customize your data queries.

* Data Extraction:

After defining your query, you can execute it to extract data from the selected sources. The extracted data can be in the form of tables or datasets, depending on the type of analysis you want to perform.



* Data Transformation:

In many data analysis scenarios, you might need to clean, transform, or reshape the extracted data. Cognos offers tools for data transformation and cleansing to ensure that the data is accurate and suitable for analysis.

* Data Integration:

If you're working with data from multiple sources, you can integrate and combine data from various sources into a unified dataset. This is especially important for comprehensive analysis.

* Data Storage:

Cognos typically doesn't store the extracted data permanently, but you can save the results of your analysis, reports, or dashboards for future reference.

* Analysis and Reporting:

Cognos typically doesn't store the extracted data permanently, but you can save the results of your analysis, reports, or dashboards for future reference

* Data Refresh and Automation:

Depending on your needs, you may want to set up automated data extraction and analysis processes to keep your data up to date. Cognos can schedule data refreshes and automate report generation

* Deployment and Sharing:

Finally, you can deploy your analysis results and share them with relevant stakeholders, which can be done through Cognos reports, dashboards, or other means.

**Analytics in Action:**

This data is often structured and organized to support data analysis and decision-making. Analytical data in Cognos can come from various sources and be transformed and modeled to meet the specific needs of business users. Here are key aspects related to analytical data in Cognos

* Data Sources:

Analytical data can originate from various sources, including databases, data warehouses, spreadsheets, and external data feeds. Cognos can connect to these sources to access and analyze data.

* Data Modeling:

Data modeling in Cognos involves defining the structure and relationships within the data. It helps users understand how different data elements are related and organized, making it easier to query and analyze the data.

* ETL (Extract, Transform, Load):

Data extraction, transformation, and loading are crucial processes to prepare data for analysis. Cognos provides tools and capabilities to extract data from source systems, transform it to suit analytical needs, and load it into its data model.

* Data Exploration:

Cognos offers features for data exploration, allowing users to navigate, filter, and interact with the data to identify patterns, trends, and anomalies.

* Data Visualization:

Cognos enables users to create various data visualizations, such as charts, graphs, and dashboards, to represent data in a visually meaningful way.

* Reporting:

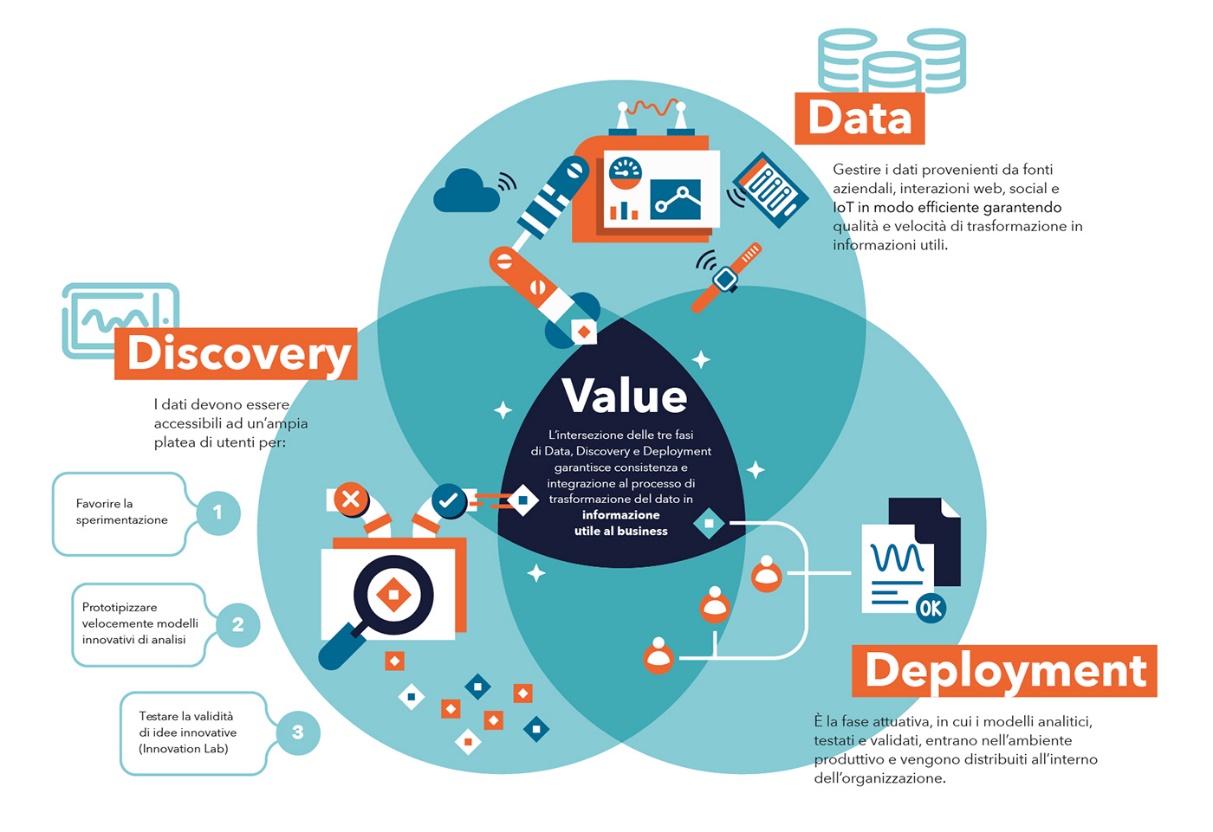
Cognos allows users to create reports based on the analytical data. These reports can be customized, scheduled, and distributed to relevant stakeholders.

* Ad-Hoc Analysis:

Business users can perform ad-hoc analysis by creating their own queries and reports, enabling them to explore data and generate insights without relying on IT or data analysts.

* Data Security and Governance:

Cognos provides tools for managing data security and governance. This ensures that only authorized users have access to sensitive data, and data remains compliant with organizational policies and regulations.



* Data Integration:

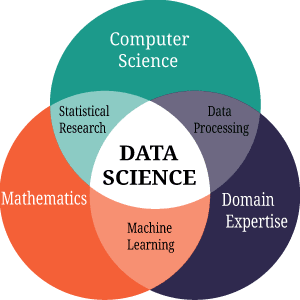
Analytical data in Cognos may involve the integration of data from various sources, making it possible to analyze and report on a comprehensive set of data.

* Data Performance:

Cognos optimizes data retrieval and analysis performance, ensuring that analytical processes run efficiently, even with large datasets.

**Python for Data Science:**

Python is an incredibly popular and versatile programming language for data science. It offers a wide range of libraries and tools that make it a powerful choice for tasks like data analysis, data visualization, machine learning, and more. Here's a brief overview of some of the key libraries and tools used in Python for data science.



**Using the Built-In Libraries (Pandas):**

Pandas is a powerful Python library for data manipulation and analysis. It provides data structures and functions for working with structured data, making it an essential tool for data scientists and analysts. Here are some of the key features and concepts related to Pandas:

* Data Structures:

DataFrame: A two-dimensional, tabular data structure with labeled axes (rows and columns). It's similar to a spreadsheet or SQL table and is the primary data structure used in Pandas.

Series: A one-dimensional labeled array capable of holding data of various types. Series can be thought of as a column in a DataFrame.

* Data Import and Export:

Pandas can read and write data from/to various file formats, including CSV, Excel, SQL databases, and more.

Common methods for reading data include pd.read\_csv(), pd.read\_excel(), and pd.read\_sql().

Data can be exported using methods like to\_csv(), to\_excel(), and to\_sql().

* Data Cleaning and Preprocessing:

Pandas provides functions for handling missing data, duplicate data, and data type conversion.

Methods like drop(), fillna(), replace(), and astype() are used for data cleaning.

* Indexing and Selection:

You can use labels or integer-based indexing to select and filter data in Pandas DataFrames.

Common selection methods include loc[] for label-based indexing and iloc[] for integer-based indexing.

* Data Manipulation:

Pandas supports a wide range of operations, such as filtering, grouping, aggregating, sorting, and merging datasets.

Functions like groupby(), agg(), sort\_values(), and merge() are commonly used for these operations.

* Data Visualization:

While Pandas itself is not primarily a visualization library, it can be easily integrated with libraries like Matplotlib and Seaborn for data visualization.

Pandas DataFrames can be used to create basic plots and charts.

* Time Series Data:

Pandas has robust support for time series data, with specialized data structures and functions for handling date and time-related operations.

The Timestamp and DatetimeIndex are used for working with dates and times.

* Reshaping Data:

Pandas provides methods for reshaping data, such as pivot, melt, and stack/unstack, which are useful for transforming data from wide to long format and vice versa.

* Handling Categorical Data:

Categorical data can be handled efficiently using Pandas' Categorical data type, which can reduce memory usage and improve performance.

* Statistical Analysis:

Pandas offers a wide range of statistical functions for basic data analysis and summary statistics.

Functions like mean(), median(), sum(), and describe() can provide insights into the data.

* Combining Data:

You can merge, join, and concatenate DataFrames to combine data from multiple sources using methods like concat(), merge(), and join().

**Analytics of Data:**

Here we use the TN-Marginal workers dataset, further analyzing the data we show some examples of visualization following.

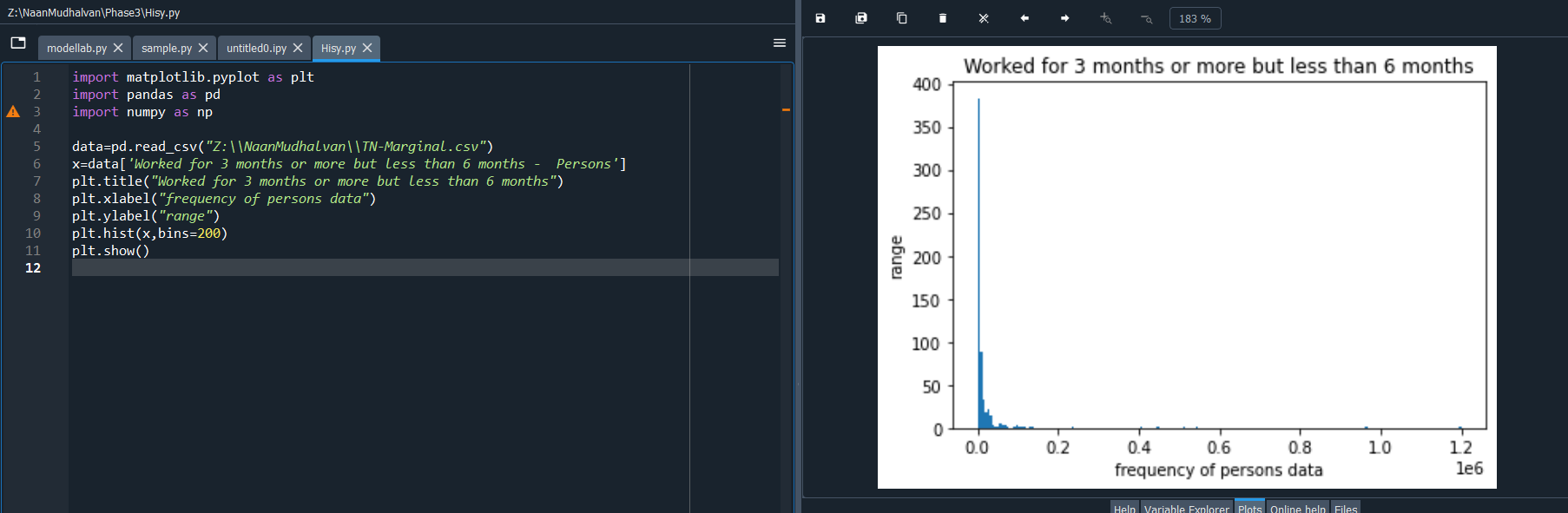
Worked Range Analysis:

That category satisfying more visualization tool, but here we choose the histogram plot,

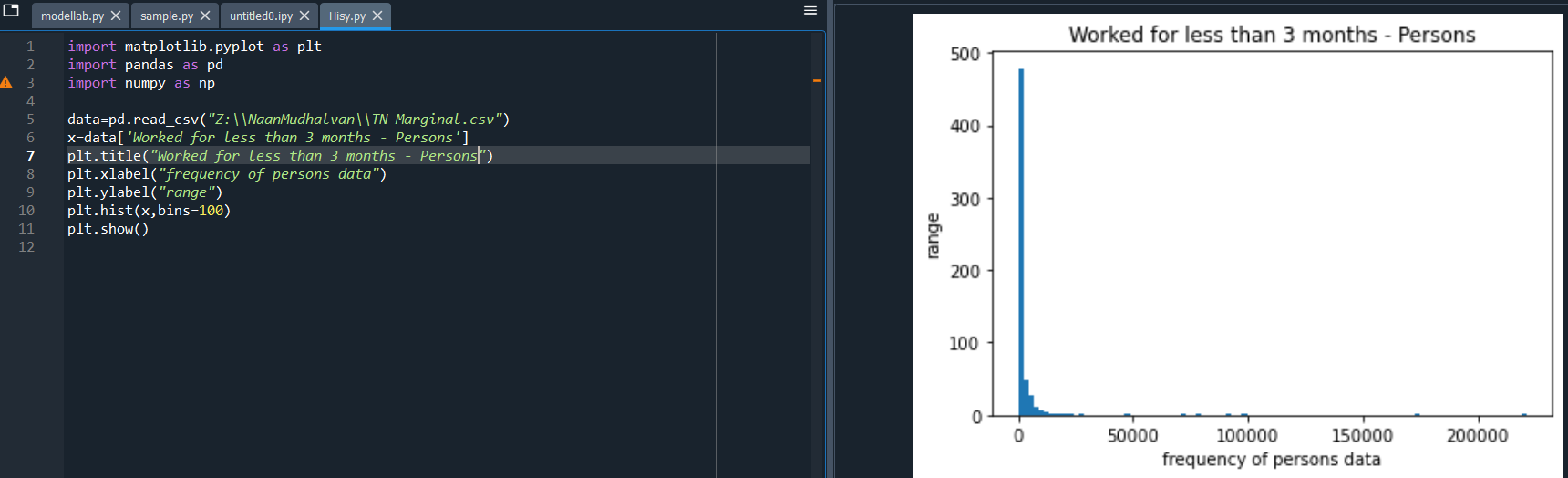
The hist is a function of matplotlib library, it require two mandatory data

1. X -data
2. Bins-range/interval

* Worked for 3 months or more but less than 6 months



* Worked for less than 3 months – Persons

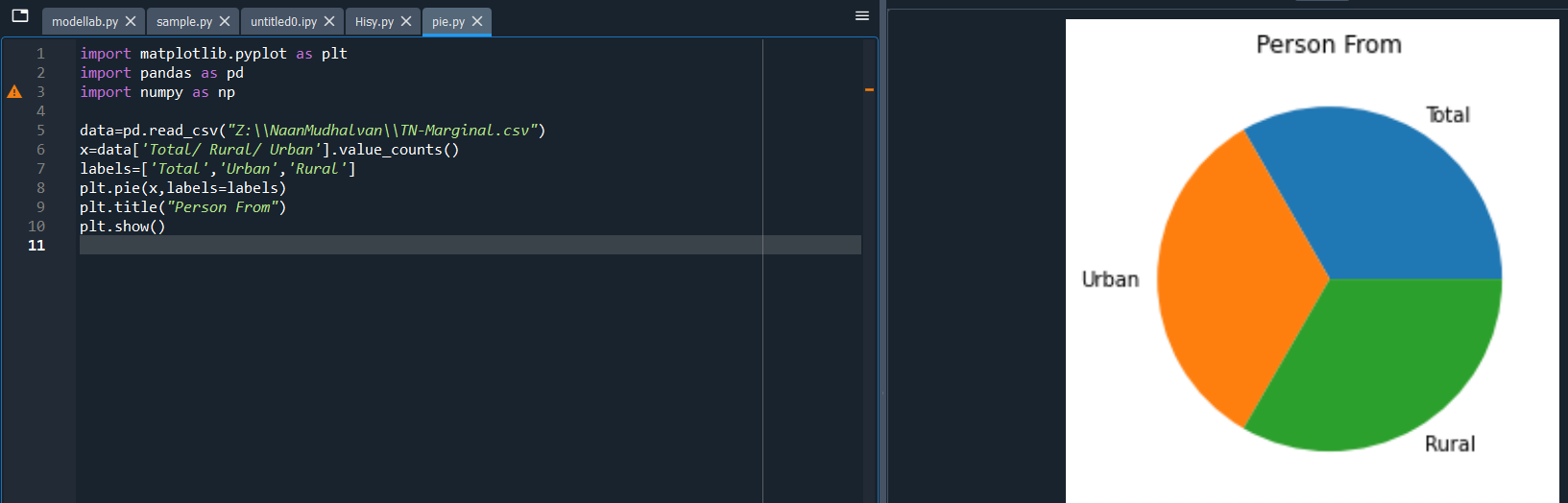


Person From:

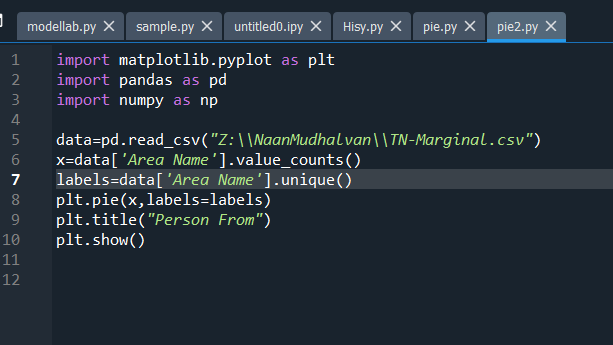
That category satisfying more visualization tool, but here we choose the Pie Chart,

The pie chart requires one mandatory parameter data.

Labels and explode are not necessary

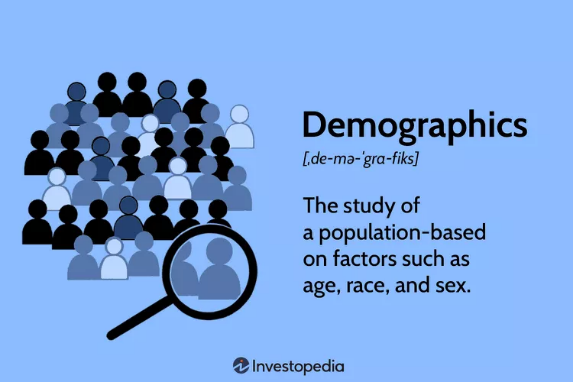
****

Person from (district vise):



**Demographic Analysis:**

Demographic analysis is a process of studying and understanding the characteristics of a population. It helps us learn more about the people who make up a particular group, community, or society. By examining various aspects such as age, gender, income, education, and more, we can gain valuable insights into a population's composition and how it is changing over time.



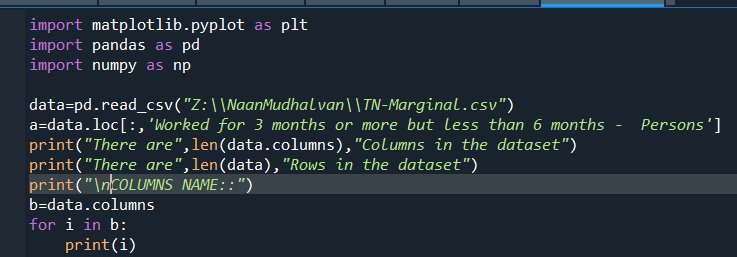
One important aspect of demographic analysis is creating visualizations. Visualizations are like pictures or graphs that represent data in a way that is easy to understand. They help us see patterns and trends within the population data. In this process, we'll use charts, graphs, and maps to illustrate the information we gather, making it simpler for everyone to grasp and interpret. This helps decision-makers, researchers, and the general public to make informed choices and understand the dynamics of a population.

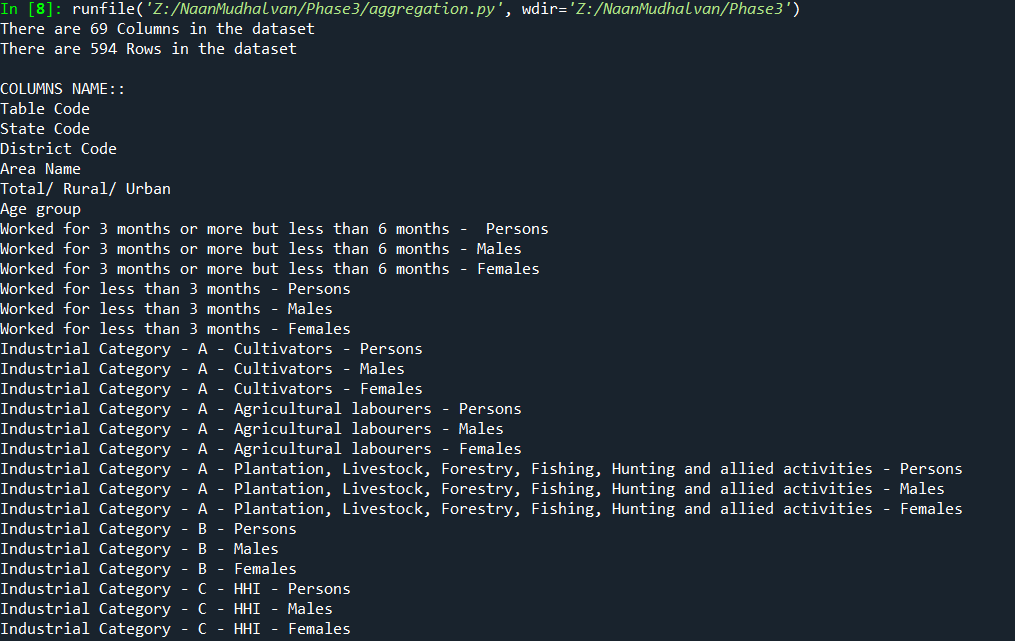
**“The Study of a Population based on factors such as age, language and sex”**

**Aggregation functions :**

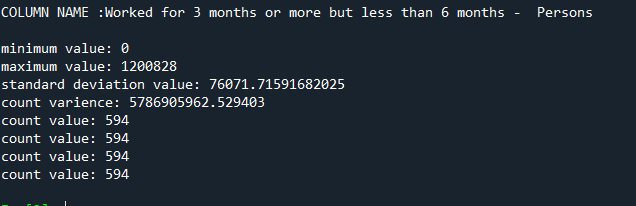
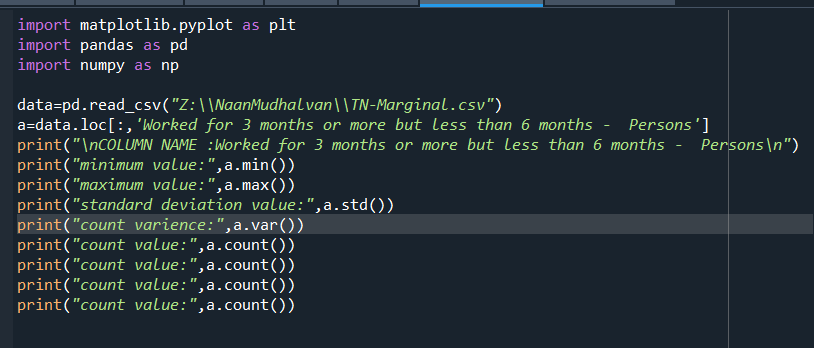
Pandas is a powerful Python library for data manipulation and analysis. It provides data structures and functions for working with structured data, making it an essential tool for data scientists and analysts. Here are some of the key features and concepts related to Pandas:

The detail about dataset:





Common aggregation functions (min , max):



Statistical analysis:

print(df.describe())

output:

Worked for 3 months or more but less than 6 months - Persons ... Industrial Category - R to U - Non HHI - Females

count 5.940000e+02 ... 594.000000

mean 1.617277e+04 ... 892.754209

std 7.607172e+04 ... 3988.125301

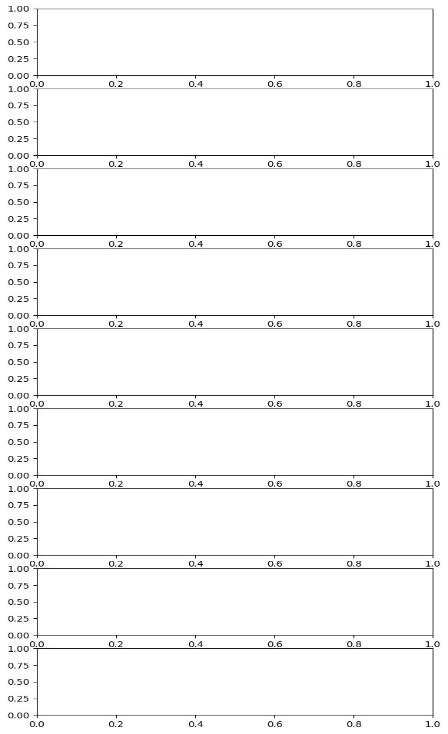
min 0.000000e+00 ... 0.000000

25% 2.872500e+02 ... 30.500000

50% 2.225500e+03 ... 135.000000

75% 9.628500e+03 ... 500.000000

max 1.200828e+06 ... 66287.000000

[8 rows x 63 columns]

Check the outliers:

fig, axs = plt.subplots(9,1,dpi=95, figsize=(7,17))

i = 0

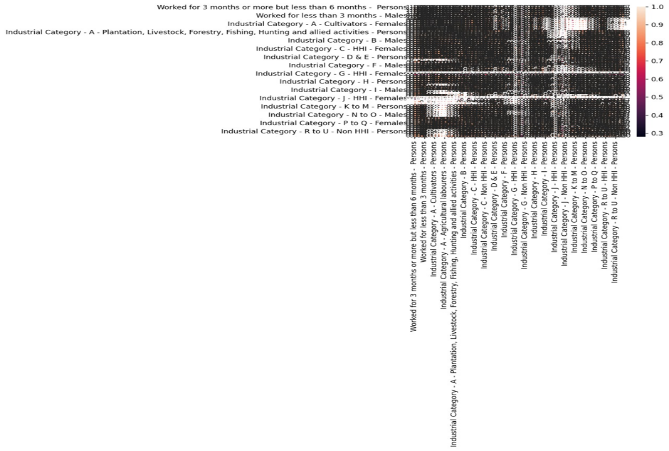
for col in df.columns:

axs[i].boxplot(df[col], vert=False)

axs[i].set\_ylabel(col)

i+=1

plt.show()



Correlation:

corr = df.corr()

plt.figure(dpi=130)

sns.heatmap(df.corr(), annot=True, fmt= '.2f')

plt.show()

Missing value:

print(df.isnull().sum())

output:

Table Code 0

State Code 0

District Code 0

Area Name 0

Total/ Rural/ Urban 0

...

Industrial Category - R to U - HHI - Males 0

Industrial Category - R to U - HHI - Females 0

Industrial Category - R to U - Non HHI - Persons 0

Industrial Category - R to U - Non HHI - Males 0

Industrial Category - R to U - Non HHI - Females 0

Length: 69, dtype: int64

**Visualization using matplotlib and seaborn:**

Here we use the TN-Marginal workers dataset, further analyzing the data we show some examples of visualization following.

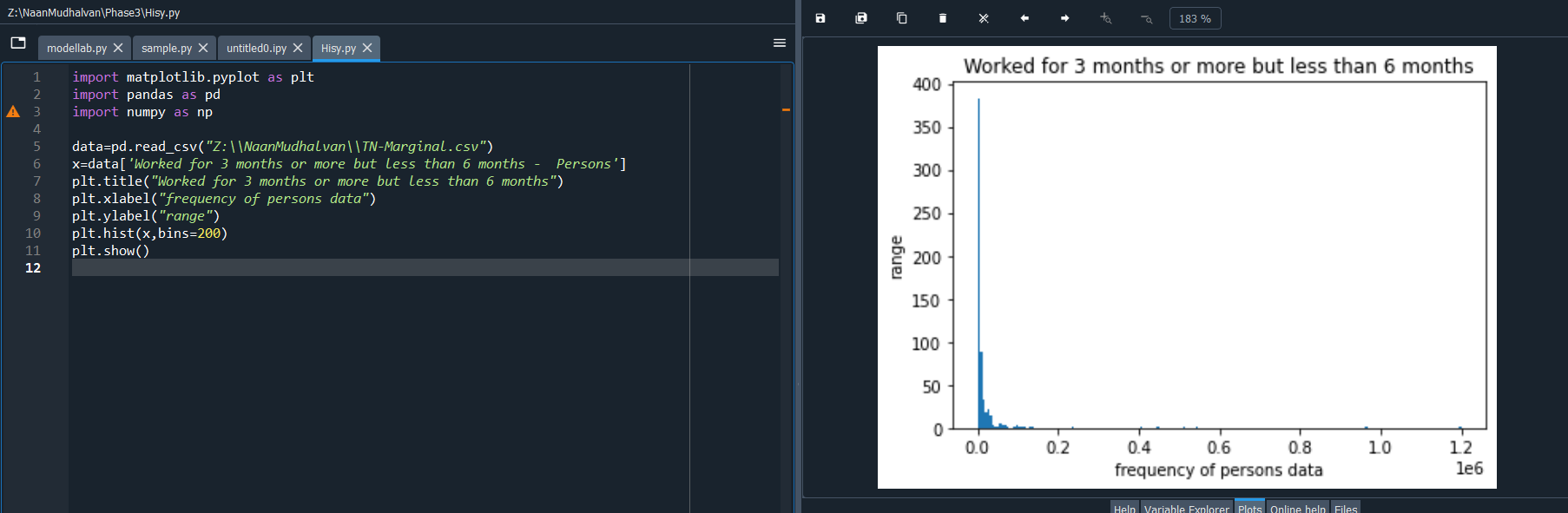
Worked Range Analysis:

That category satisfying more visualization tool, but here we choose the histogram plot,

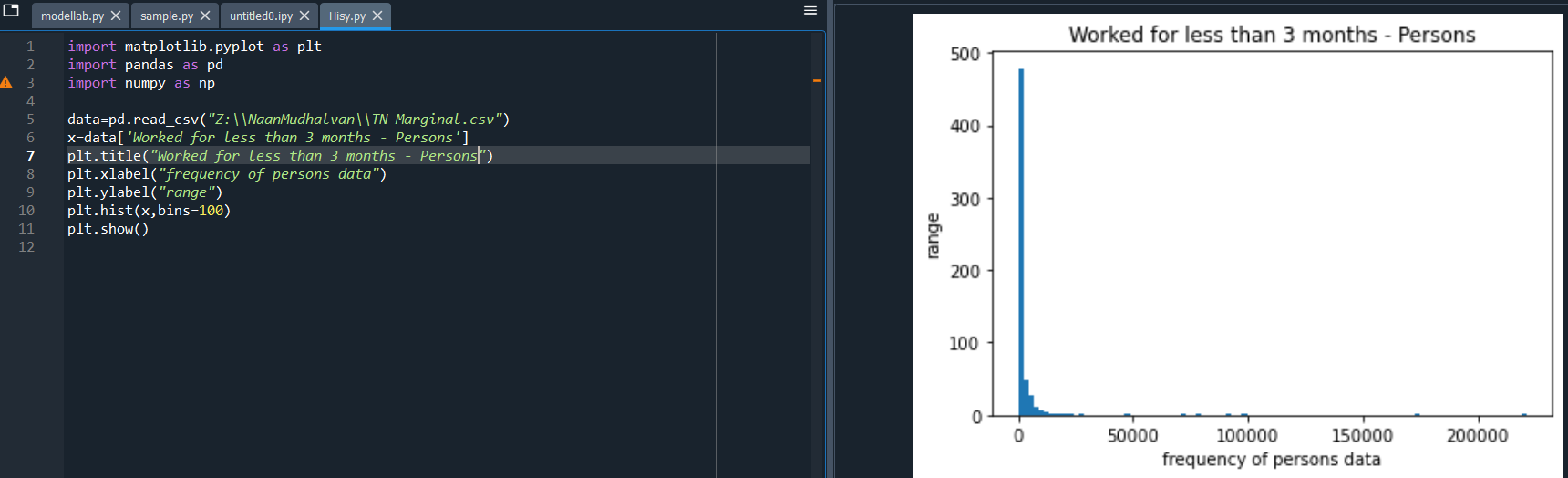
The hist is a function of matplotlib library, it require two mandatory data

1. X -data
2. Bins-range/interval

* Worked for 3 months or more but less than 6 months



* Worked for less than 3 months – Persons

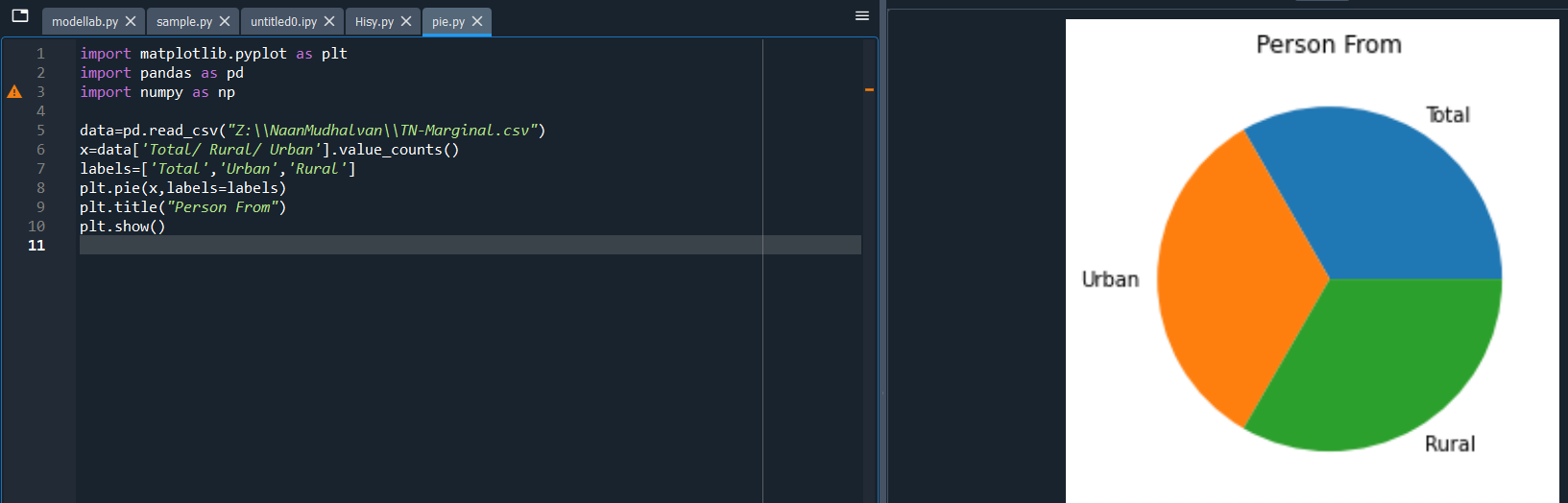


Person From:

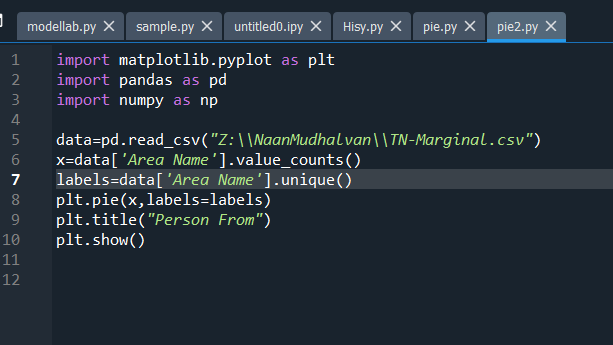
That category satisfying more visualization tool, but here we choose the Pie Chart,

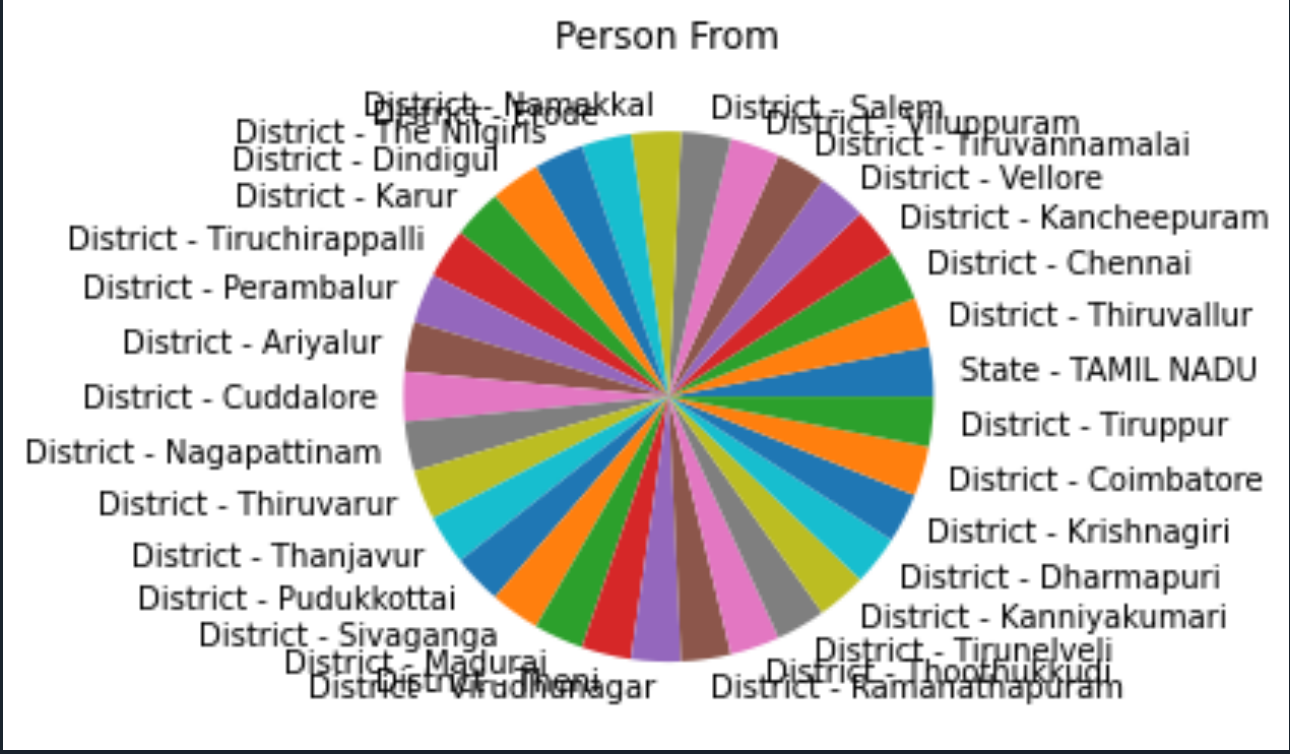
The pie chart requires one mandatory parameter data.

Labels and explode are not necessary

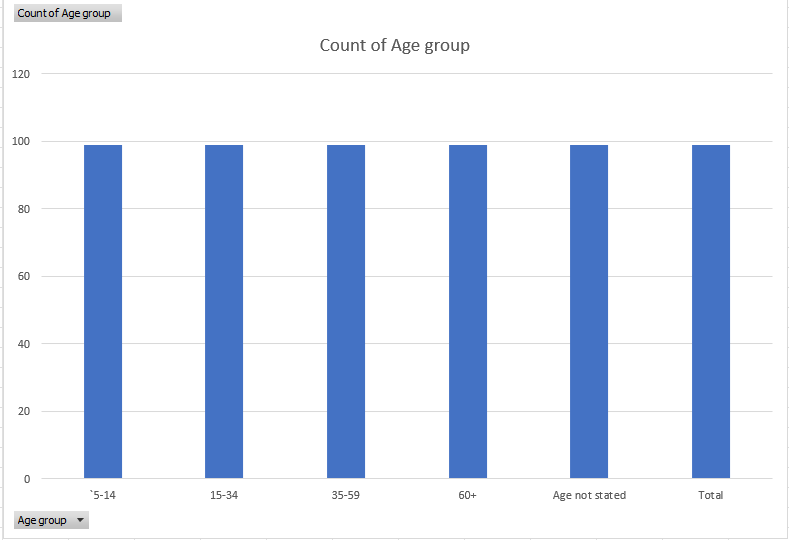
****

Person from (district vise):

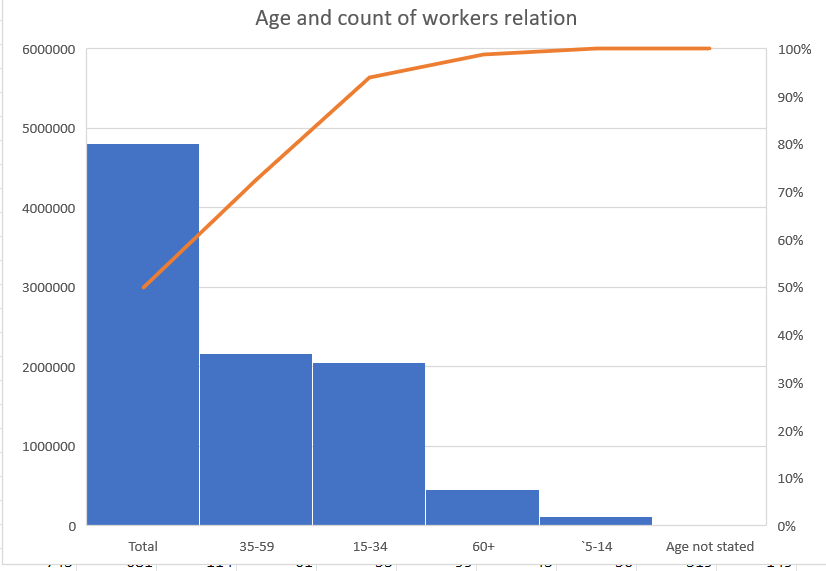


****

* frequency of age-group

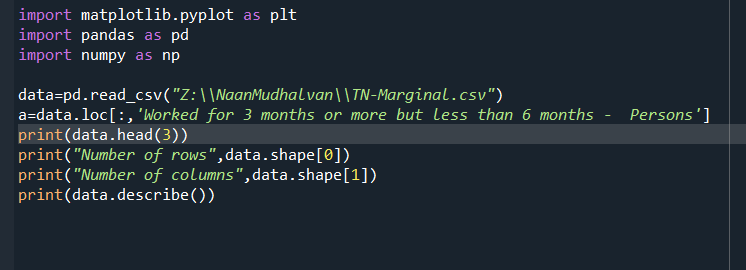
****

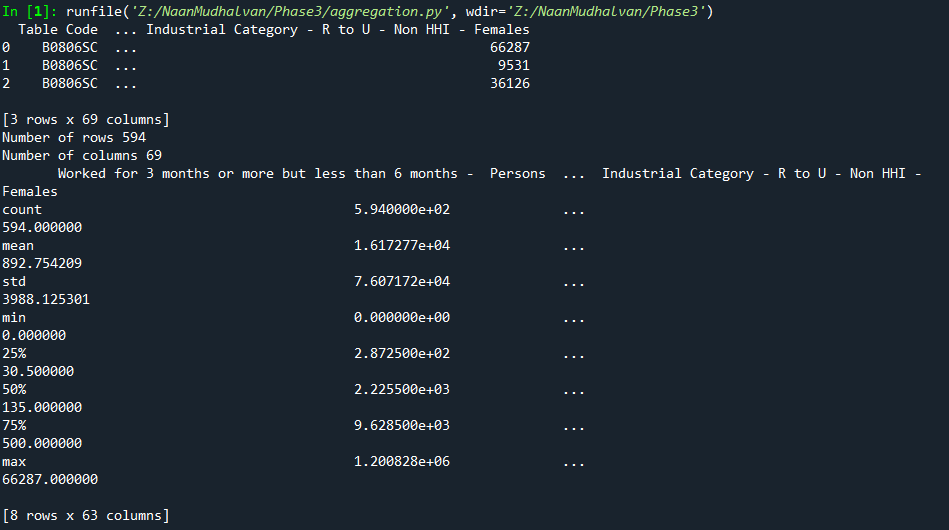
* Age and frequent of person worked within 6 months

****

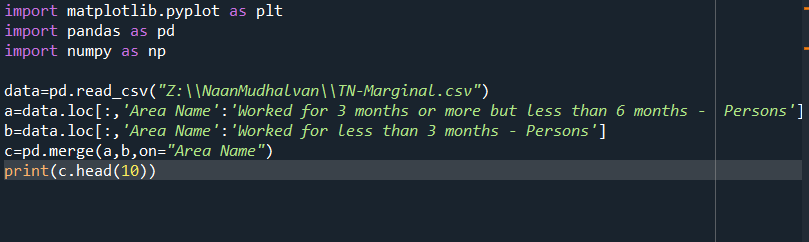
**Data Manipulation:**

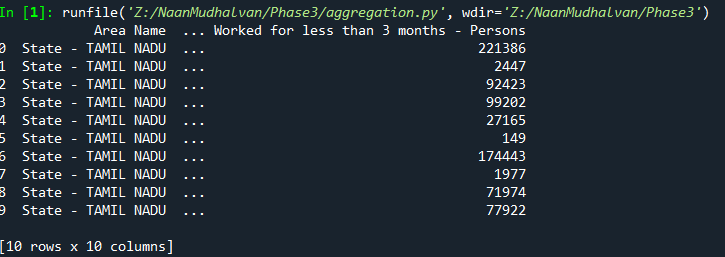
The data in the real world is very unpleasant & unordered so by performing certain operations we can make data understandable based on one’s requirements, this process of converting unordered data into meaningful information can be done by data manipulation.

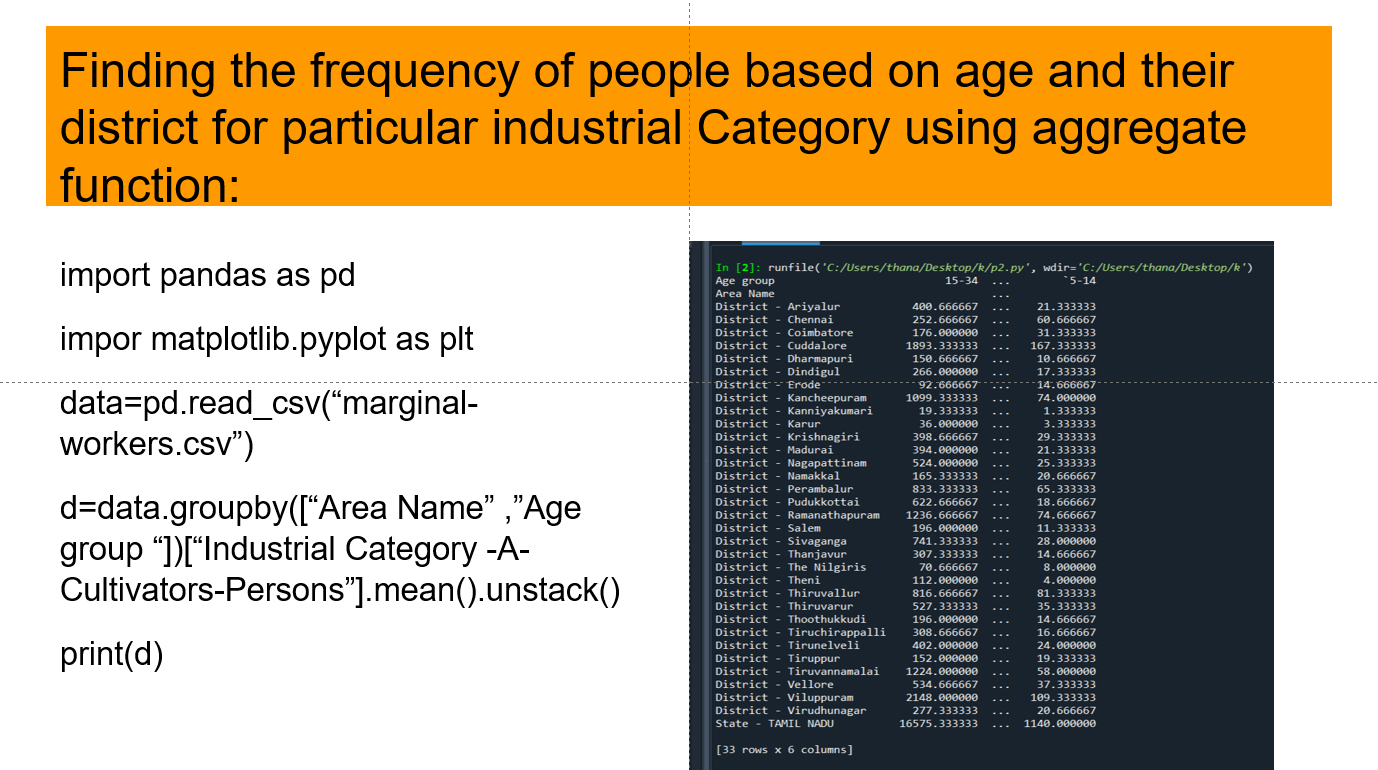
****

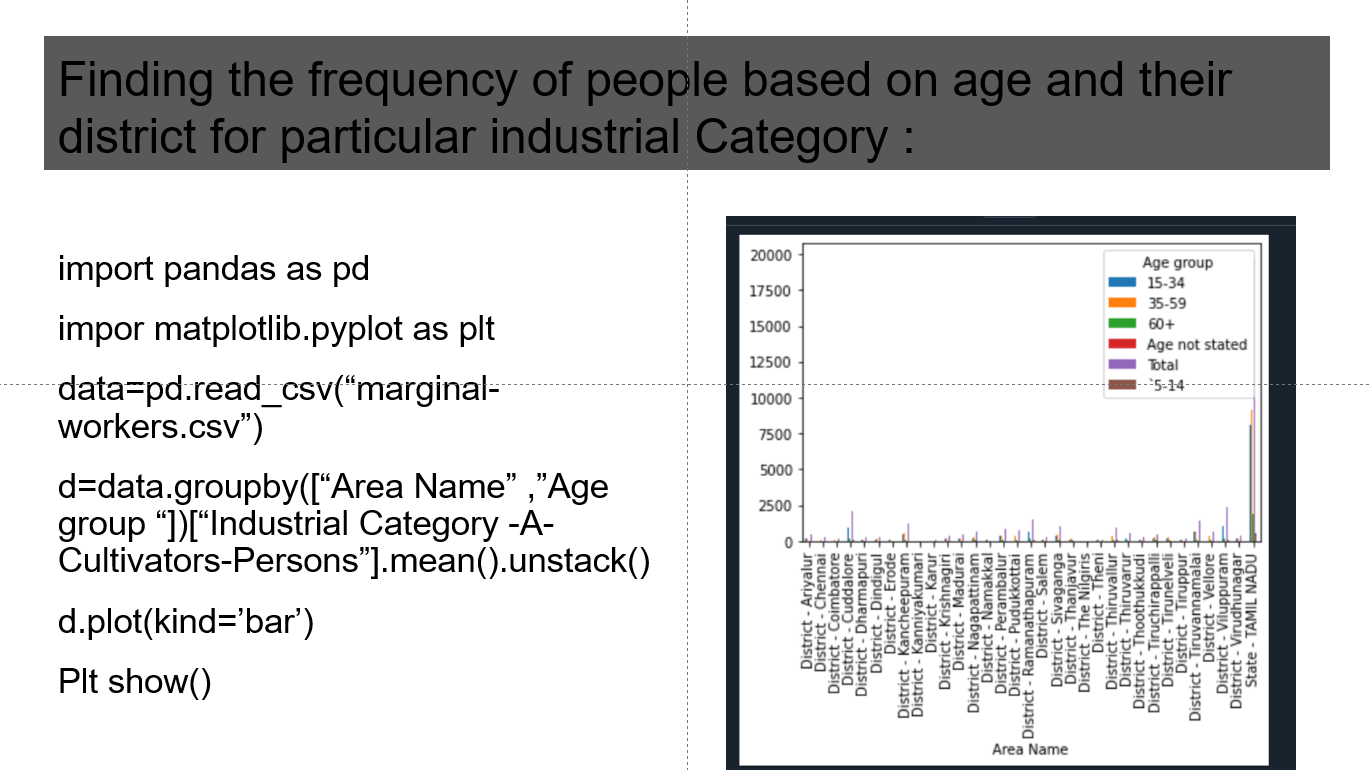
****

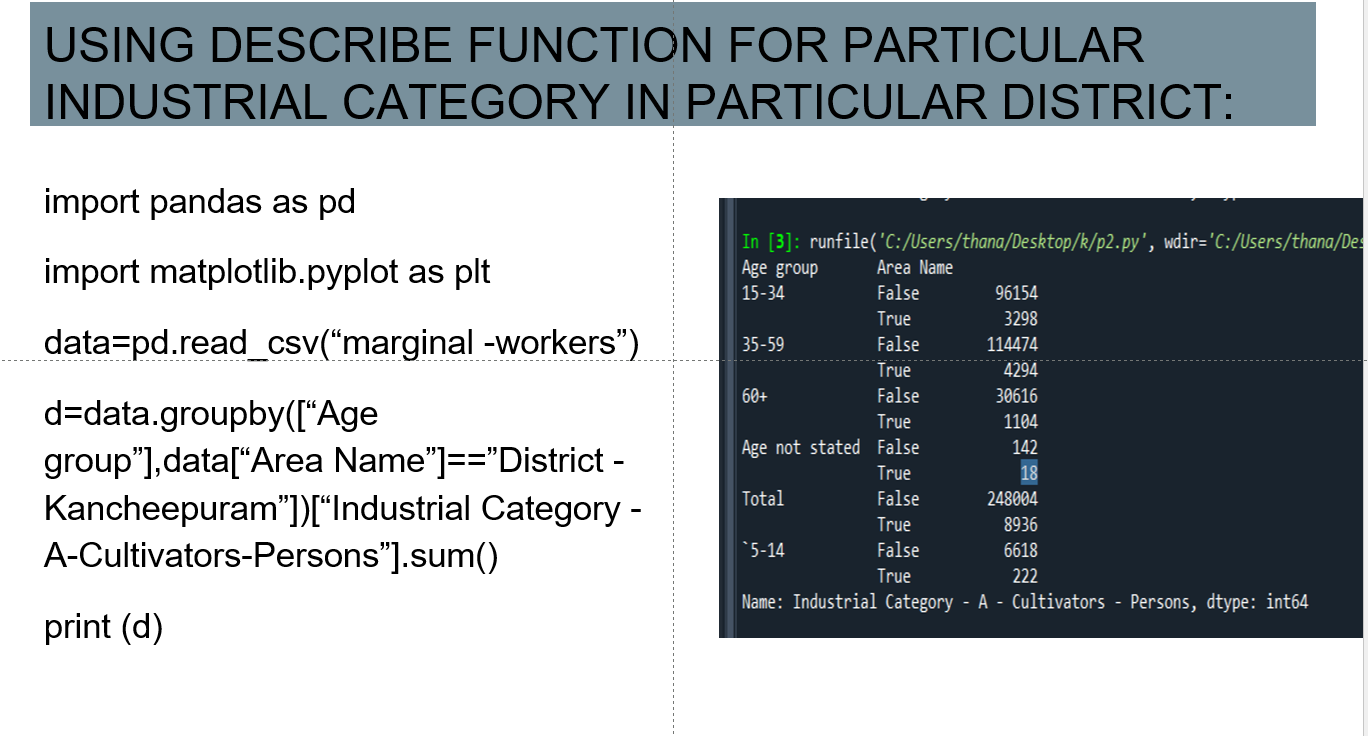
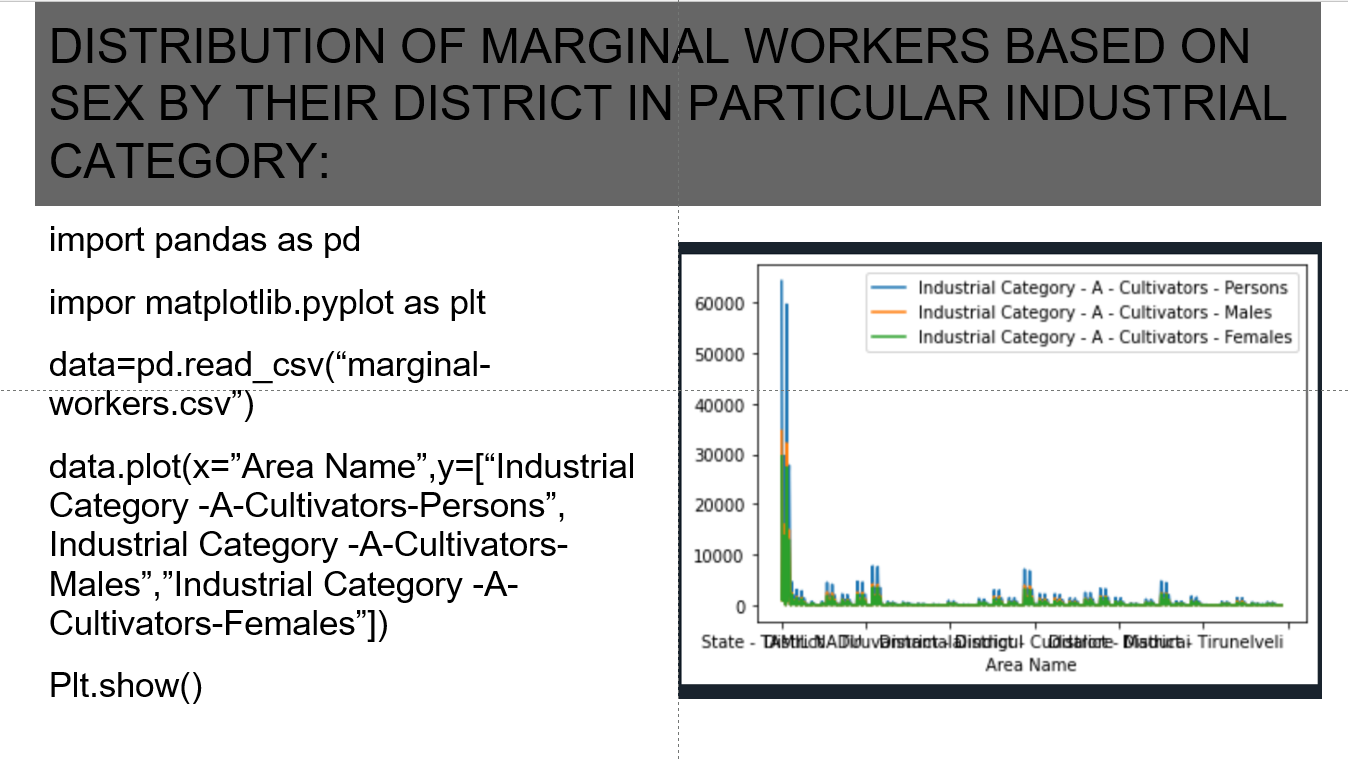
Merge function:

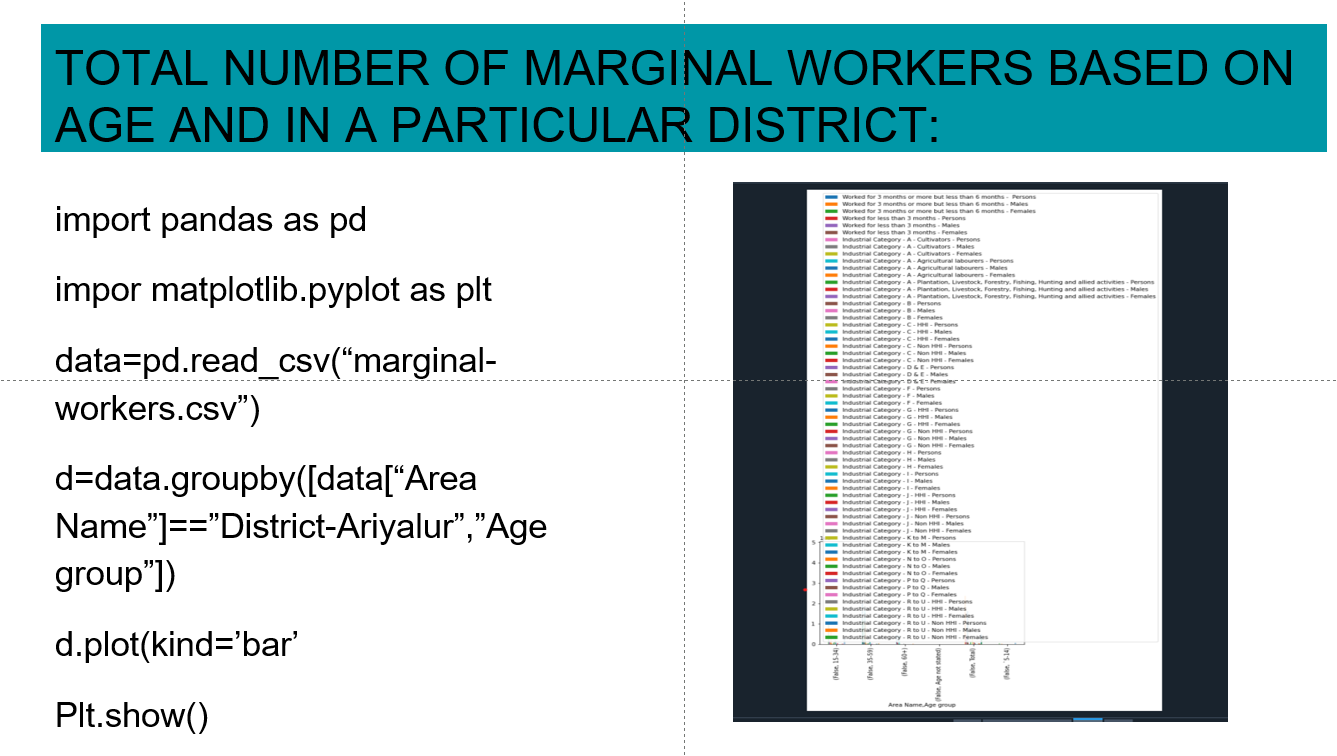
****

****

****

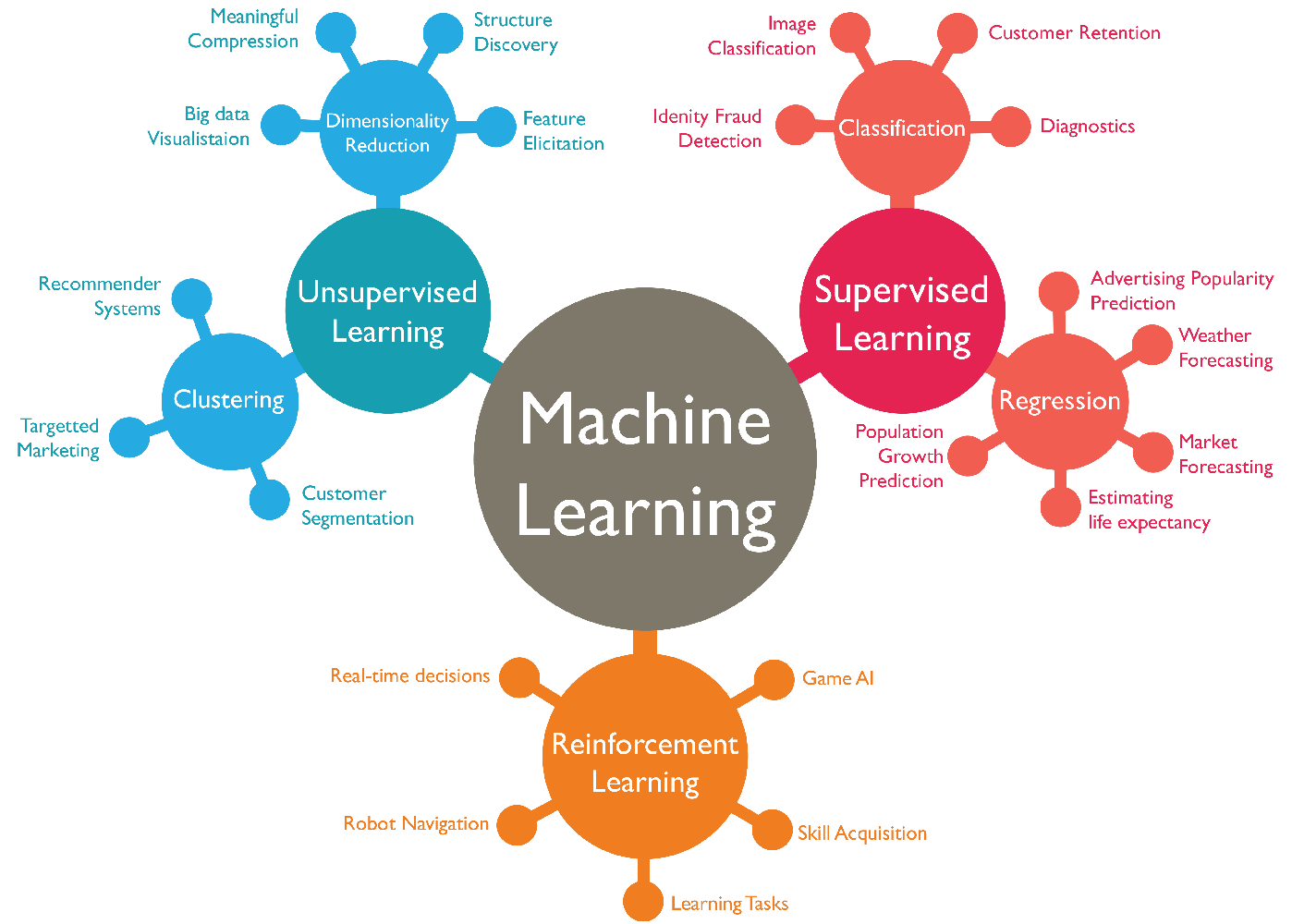
****

****

****

**Machine-Learning**

Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.



Machine Learning techniques

* Regression/Estimation
* Classification
* Clustering
* Associations
* Anomaly Detection
* Sequence Mining
* Dimension Reduction
* Recommendation Systems

Python Libraries for ML:

|  |  |  |  |
| --- | --- | --- | --- |
| NumPy: Cómo ordenar una matriz NumPy en Python - Analytics Lane | Data Science Academy Mexico | Matplotlib logo — Matplotlib 3.1.0 documentation | What is Pandas? — Geo-Python 2017 Autumn documentation |



*Scikit-learn* is both, well-documented and straightforward to learn/use if you want an introduction to machine learning, or if you want the most up-to-date ML testing tool. It lets you construct a predictive data model with a few lines of code and then apply that model to your data as a high-level library. It’s flexible and integrates nicely with other Python libraries such as Matplotlib for charts, Numpy for numerical computations, and Pandas for DataFrames.

Scikit-learn contains many supervised & unsupervised learning algos. Most importantly, it is by far the simplest and cleanest ML library. It was created with a software engineer’s perspective. Its central API architecture revolves around being simple to use while still being versatile and flexible for research endeavors. Because of its robustness, it is suitable for use in any end-to-end ML project — from research to production deployments. It is based on the machine learning libraries mentioned below:

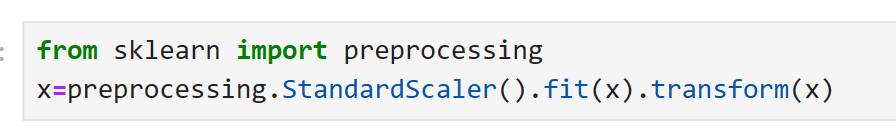
*NumPy*: is a Python library that allows you to manipulate multidimensional arrays and matrices. It also includes a large set of mathematical functions for performing various calculations

*SciPy*: is an environment of libraries for performing technical programming tasks

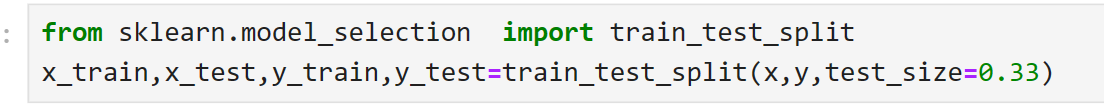
*Matplotlib*: is a library that can be used to build different charts and graphs

Sci-kit learn

Example:



Here, preprocessing package provides several common utility functions and transformer classes to change.

****

You have to split your dataset into train and sets to train your model and then test models’ accuracy separately.

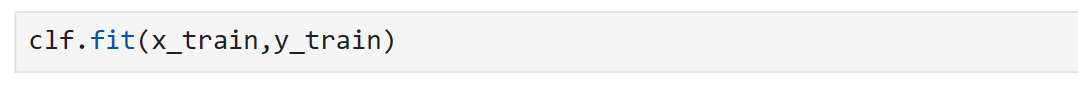
Scikit learn can split arrays or matrices into random train & test subsets.

Then you setup your algorithm,

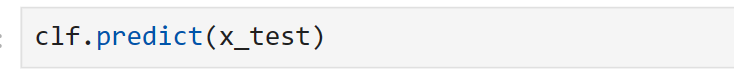
Eq: SVM



Train your model,

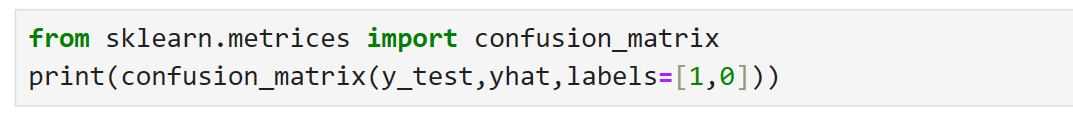
****

Use test set to predications,

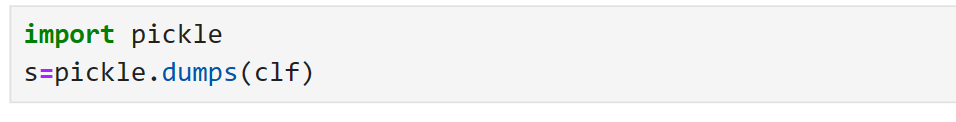
****

Also you can use different metrices to evaluate your model occurency

Example, using confusion matrices



finally save your model,



Supervise learning:

Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output.





1. Regression

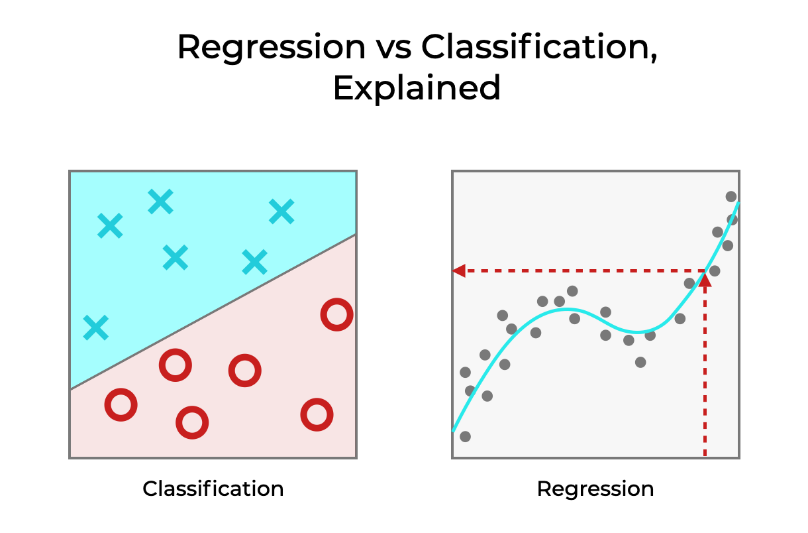
Regression algorithms are used if there is a relationship between the input variable and the output variable. It is used for the prediction of continuous variables, such as Weather forecasting, Market Trends, etc. Below are some popular Regression algorithms which come under supervised learning:

* Linear Regression
* Regression Trees
* Non-Linear Regression
* Bayesian Linear Regression
* Polynomial Regression

2. Classification

Classification algorithms are used when the output variable is categorical, which means there are two classes such as Yes-No, Male-Female, True-false, etc.

* Spam Filtering,
* Random Forest
* Decision Trees
* Logistic Regression
* Support vector Machines



**Conclusion:**

In conclusion, the "Data Analytics with Cognos" course has been a transformative experience. It has armed me with the knowledge and skills needed to navigate the data-rich landscape of today. Data analytics with Cognos is not just a course; it is a gateway to unlocking the power of data.

