FACULTY OF ENGINEERING AND TECHNOLOGY

A REPORT ABOUT MATRIX LABARATORY DATA PROCESSING AND EXPORT WORK FLOW OF ASSIGNMENT ONE AND TWO

COURSE UNIT; COMPUTER PROGRAMMING

COURSES; AMI, APE, WAR, MEB

GROUP 7

LECTURER: MR.MASERUKA BENDICTO

DECLARATION

We declare that this information in this report is our own, to the best of our knowledge.

NAME	COURSE	REG NUMBER	SIGN
KAKURU OBADIAH	WAR	BU/UP/2024/1029	
SSEBAKIJJE HOSEA	AMI	BU/UP/2024/0847	
SEMBERA SHERINA TAPINESS	MEB	BU/UP/2024/5349	
ADIPO HOPE ODWARE	MEB	BU/UP/2024/0980	
KOBUSINGYE BETHLINE	PTI	BU/UG/2024/2602	
NYERO MARIA ASUMPTA	WAR	BU/UP/2024/1053	
GUDOI ALLAN	WAR	BU/UP/2024/4449	
ETYANG JONATHAN	APE	BU/UG/2024/2671	
ESARAIT BRIAN	MEB	BU/UP/2024/5343	

DATE OF SUBMISSION: ………………………………

SUBMITTED TO: ………………………………

SIGNATURE: ……………………………………………..

APPROVAL

We are presenting this report which has been written and produced under our efforts. We carried out research on converting tables into structural arrays, outputting variables into single workbooks and writing codes that can store affirmation attributes.

ACKNOWLEDGEMENT

First and foremost, we would like to thank the Almighty God for giving us the knowledge and guidance while doing our assignment as group 7. We extend our gratitude to all the persons with whose help we managed to make it this far

The the love of every group member to invest time and provide all they could to see the assignment a success. Finally, we would like to express our gratitude to all the sources and references that have been cited in this report

DEDICATION.

We dedicate this report to all Group 7 members, who have been there with us in the process of researching and doing the and compiling this report.   To our lovers and To our lecturer Mr. Maseruka Benedicto whose guidance and expertise have been so needful, your mentorship and lecturing has built our understanding.

LIST OF ACRONYMS/ ABBREVIATIONS.

1. GUI - Graphical user interface.
2. MATLAB - Matrix Laboratory.

ABSTRACT

We started our first meeting for research on 15th, September, 2025 in the university library (room 3) out of which we were exposed to various concepts on how to interact with the matlab interface, import, extract and feeding in data in to MATLAB as per the assignment which consisted of retrieving excel data from Kaggle.com website ,copying variables of each year to tables, converting tables to structural arrays, outputting each variable in to a single work book and generating a MATLAB code that can store each members' affirmation attributes, we managed to achieve this through group work and division of tasks

Table of Contents

1.0 CHAPTER ONE INTRODUCTION

1.1 HISTORICAL BACKGROUND

MATLAB, which stands for matrix laboratory, is a high-performance programming language and environment designed primarily for technical computing. Its origins trace back to the late 1970s when Cleve Moler, a professor of computer science, developed it to provide his students with easy access to mathematical software libraries without requiring them to learn Fortran.

MATLAB is built around the concept of matrices, making it particularly effective for linear algebra and matrix manipulation. It provides a vast library of built-in functions for mathematical operations, statistics, optimization, and other specialized tasks.

MATLAB offers powerful tools for creating 2D and 3D plots, enabling users to visualize data effectively. Specialized toolboxes extend MATLAB's capabilities, providing functions tailored for specific applications like signal processing, image processing, control systems, and machine learning.

MATLAB can interface with other programming languages (like C, C++, and Python) and software tools, allowing for flexible integration into larger systems. Its interactive environment features a command window, workspace, and editor, making it accessible for both beginners and advanced users.

1.2Historical Background

The first version of MATLAB was created in Fortran in the late 1970s as a simple interactive matrix calculator. This early iteration included basic matrix operations and was built on top of two significant mathematical libraries: LINPACK and EISPACK, which were developed for numerical linear algebra and eigenvalue problems, respectively.

Recent versions of MATLAB have introduced features like the *Live Editor*, which allows users to create interactive documents that combine code, output, and formatted text. This evolution reflects MATLAB's ongoing adaptation to meet the needs of its diverse user base across academia and industry.

1.3STUDY METHODOLOGY

1.4 Introduction

At the start, each member was given a task of making research about the assignment before our first meeting. The research concepts were obtained through watching tutorials on youtube and also consultations from other continuing students especially those in year three and four.

2.0 CHAPTER 2 NUMBER ONE

There is a website on the internet called kaggle.com each group should be able to retrieve a unique data set in excel format. The group will read this data set into Mat lab in one code, they will be able to copy variables of each year and put them in the following;

1. Tables for each year of data

2. Convert the tables in 1 above into structural arrays

3. Output each of the variables in 2 above into a single workbook with each year on separate sheets having clear column headings sheet names

2.1 STEPS INVOLVED

Step1; Open Kaggle.com website from Google chrome.

Step2; sign in Kaggle.com and download any desired excel dataset and zip it in to a desired folder. The data we downloaded from kaggle.com was about the economic performance of different countries.

Step 3; Open the file and copy the link path into MATLAB.

Step4; go to Home tab and click on new script to open the editor.

Step 5; save the script. And save it in the directory that is in your MATLAB path or current working directory.

Step 6; reading the table to matlab using the code below

```
 MS = readtable("C:\Users\sseba\Desktop\assignment two\data set assignment
one.xlsx");
 disp(MS)
```

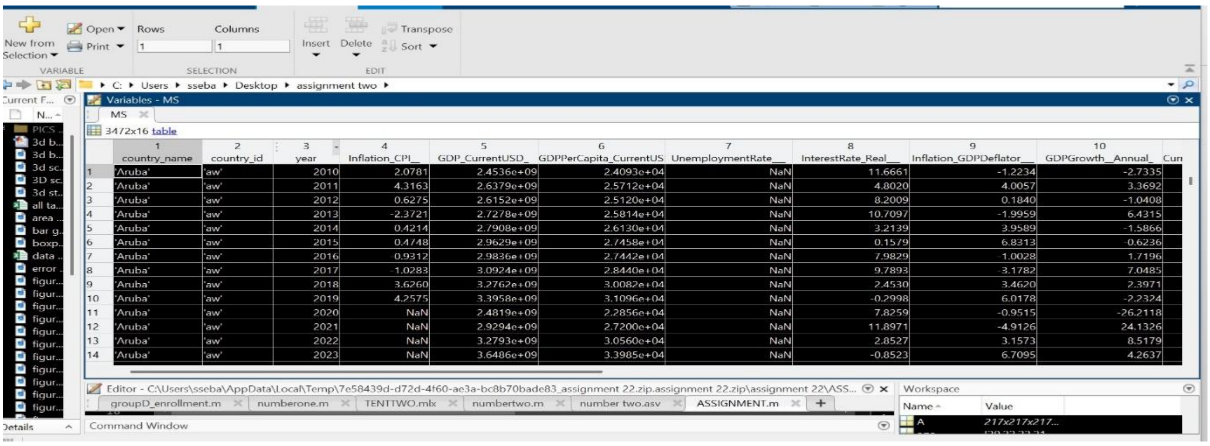
readtable(): Reads Excel data into MATLAB table format

disp(MS): Displays table contents for verification

Data Structure: Contains country economic data with columns like year, country_name, UnemploymentRate, GDP, etc

NOTE ; MS is the variable name of the table

Illustration of the table after reading into Mat lab



	1 country_name	2 country_id	3 year	4 Inflation_CPI__	5 GDP_CurrentUSD_	6 GDPPerCapita_CurrentUS	7 UnemploymentRate__	8 InterestRate_Real__	9 Inflation_GDPDeflator__	10 GDPGrowth__Annual_
1	'Aruba'	'aw'	2010	2.0781	2.4536e+09	2.4093e+04	NaN	11.6661	-1.2234	-2.7335
2	'Aruba'	'aw'	2011	4.3163	2.6379e+09	2.5712e+04	NaN	4.8020	4.0057	3.3692
3	'Aruba'	'aw'	2012	0.6275	2.6152e+09	2.5120e+04	NaN	8.2009	0.1840	-1.0408
4	'Aruba'	'aw'	2013	-2.3721	2.7278e+09	2.5814e+04	NaN	10.7097	-1.9959	6.4315
5	'Aruba'	'aw'	2014	0.4214	2.7908e+09	2.6130e+04	NaN	3.2139	3.9589	-1.5866
6	'Aruba'	'aw'	2015	0.4748	2.9629e+09	2.7458e+04	NaN	0.1579	6.8313	-0.6236
7	'Aruba'	'aw'	2016	-0.9312	2.9836e+09	2.7442e+04	NaN	7.9829	-1.0028	1.7196
8	'Aruba'	'aw'	2017	-1.0283	3.0924e+09	2.8440e+04	NaN	9.7893	-3.1782	7.0485
9	'Aruba'	'aw'	2018	3.6260	3.2762e+09	3.0082e+04	NaN	2.4530	3.4620	2.3971
10	'Aruba'	'aw'	2019	4.2575	3.3958e+09	3.1096e+04	NaN	-0.2998	6.0178	-2.2324
11	'Aruba'	'aw'	2020	NaN	2.4819e+09	2.2856e+04	NaN	7.8259	-0.9515	-26.2118
12	'Aruba'	'aw'	2021	NaN	2.9294e+09	2.7200e+04	NaN	11.8971	-4.9126	24.1326
13	'Aruba'	'aw'	2022	NaN	3.2793e+09	3.0560e+04	NaN	2.8527	3.1573	8.5179
14	'Aruba'	'aw'	2023	NaN	3.6486e+09	3.3985e+04	NaN	-0.8523	6.7095	4.2637

The second part required to separet the dataset from each year into different tables

```
MS2010 = MS(MS.year==2010, : );
MS2011 = MS(MS.year==2011, : );
% ... continues through 2025
```

The colon (:) means "include all columns.

The percentage sign '%' was used to comment in the code

Logic: Filters main table MS by year condition

MS.year==2010: Creates logical index for rows where year equals 2010

Result: 16 separate tables (2010-2025), each containing data for that specific year

MS 2010 is the variable name for the specific year

```
% separating data from each year into diffrent tables
MS2010 =MS(MS.year==2010, : );%extraction of unique years(2010)
MS2011 =MS(MS.year==2011, : );%extraction of unique years(2011)
MS2012 =MS(MS.year==2012, : );%extraction of unique years(2012)
MS2013 =MS(MS.year==2013, : );%extraction of unique years(2013)
MS2014 =MS(MS.year==2014, : );%extraction of unique years(2014)
MS2015 =MS(MS.year==2015, : );%extraction of unique years(2015)
MS2016 =MS(MS.year==2016, : );%extraction of unique years(2016)
MS2017 =MS(MS.year==2017, : );%extraction of unique years(2017)
MS2018 =MS(MS.year==2018, : );%extraction of unique years(2018)
MS2019 =MS(MS.year==2019, : );%extraction of unique years(2019)
MS2020 =MS(MS.year==2020, : );%extraction of unique years(2020)
MS2021 =MS(MS.year==2021, : );%extraction of unique years(2021)
MS2022 =MS(MS.year==2022, : );%extraction of unique years(2022)
MS2023 =MS(MS.year==2023, : );%extraction of unique years(2023)
MS2024 =MS(MS.year==2024, : );%extraction of unique years(2024)
MS2025 =MS(MS.year==2025, : );%extraction of unique years(2025)
```

The code was to check the column having a variable name Year and choose that specific year and display it separately from other years

2.2 Part (ii) of question one

Tables are MATLAB data structures that store column-oriented data with variable names

table2struct(): Converts table to structure array

Tables: Column-based, ideal for spreadsheet-like data

Structures: Field-based, better for object-oriented data

Each structure element represents one row of the original table

```
%conversion of tables into struts
S2010=table2struct(MS2010);%for MS2010
S2011=table2struct(MS2011);%for MS2011
S2012=table2struct(MS2012);%for MS2012
S2013=table2struct(MS2013);%for MS2013
S2014=table2struct(MS2014);%for MS2014
S2015=table2struct(MS2015);%for MS2015
S2016=table2struct(MS2016);%for MS2016
S2017=table2struct(MS2017);%for MS2017
```

```
S2018=table2struct(MS2024);%for MS2018
S2019=table2struct(MS2024);%for MS2019
S2020=table2struct(MS2024);%for MS2020
S2021=table2struct(MS2024);%for MS2021
S2022=table2struct(MS2024);%for MS2022
S2023=table2struct(MS2024);%for MS2023
S2024=table2struct(MS2024);%for MS2024
S2025=table2struct(MS2024);%for MS2025
```

In this code the we commanded mat lab to convert the above tables into structural arrays using x=table2struct (valuable of the table)

2.3 Part (iii)

The code below we first defined the variable workbook to equal to a file path of an empty excel work book called all tables number one

Then used the code writeable

('table valuable, location, 'sheet','name of sheet')

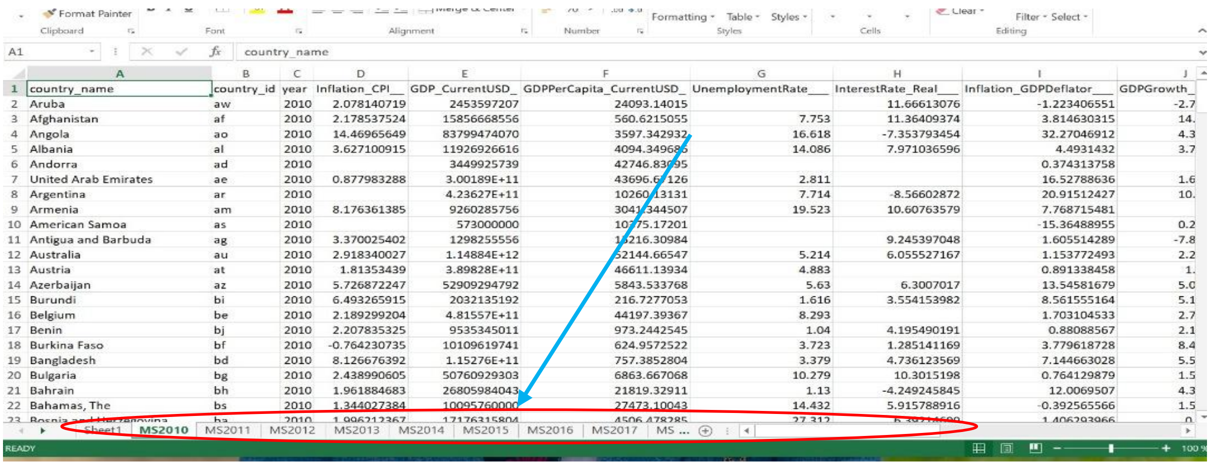
that is to say; creates a file path pointing to an Excel file called "all tables number one.xls"

```
 workbook = "C:\Users\sseba\Desktop\assignment two\all tables number
one.xlsx";
 If the file doesn't exist, it will be created. If the file exists, it will
be overwritten
 %writing each table on adiffrent sheets
 writetable(MS2010,workbook,'sheet','MS2010');
 writetable(MS2011,workbook,'sheet','MS2011');
 writetable(MS2012,workbook,'sheet','MS2012');
 writetable(MS2013,workbook,'sheet','MS2013');
 writetable(MS2014,workbook,'sheet','MS2014');
 writetable(MS2015,workbook,'sheet','MS2015');
 writetable(MS2016,workbook,'sheet','MS2016');
 writetable(MS2017,workbook,'sheet','MS2017');
 writetable(MS2018,workbook,'sheet','MS2018');
 writetable(MS2019,workbook,'sheet','MS2019');
 writetable(MS2020,workbook,'sheet','MS2020');
 writetable(MS2021,workbook,'sheet','MS2021');
 writetable(MS2022,workbook,'sheet','MS2022');
 writetable(MS2023,workbook,'sheet','MS2023');
 writetable(MS2024,workbook,'sheet','MS2024');
 writetable(MS2025,workbook,'sheet','MS2025');
```

writetable(): Exports each yearly table to separate Excel sheet

Organization: Each year gets its own named sheet in the same workbook

And the code is run to produce the table in the illustration below



	country_name	country_id	year	Inflation_CPI__	GDP_CurrentUSD_	GDPPerCapita_CurrentUSD_	UnemploymentRate___	InterestRate_Real___	Inflation_GDPDeflator___	GDPGrowth_
2	Aruba	aw	2010	2.078140719	2453597207	24093.14015		11.66613076	-1.223406551	-2.7
3	Afghanistan	af	2010	2.178537524	15856668556	560.6215055	7.753	11.36409374	3.814630315	14.
4	Angola	ao	2010	14.46965649	83799474070	3597.342932	16.618	-7.353793454	32.27046912	4.3
5	Albania	al	2010	3.627100915	11926926616	4094.349686	14.086	7.971036596	4.4931432	3.7
6	Andorra	ad	2010		3449925739	42746.83095			0.374313758	
7	United Arab Emirates	ae	2010	0.877983288	3.00189E+11	43696.67126	2.811		16.52788636	1.6
8	Argentina	ar	2010		4.23627E+11	10260.13131	7.714	-8.56602872	20.91512427	10.
9	Armenia	am	2010	8.176361385	9260285756	3041.344507	19.523	10.60763579	7.768715481	
10	American Samoa	as	2010		573000000	10275.17201			-15.36488955	0.2
11	Antigua and Barbuda	ag	2010	3.370025402	1298255556	15216.30984		9.245397048	1.605514289	-7.8
12	Australia	au	2010	2.918340027	1.14884E+12	52144.66547	5.214	6.055527167	1.153772493	2.2
13	Austria	at	2010	1.81353439	3.89828E+11	46611.13934	4.883		0.891338458	1.
14	Azerbaijan	az	2010	5.726872247	52909294792	5843.533768	5.63	6.3007017	13.54581679	5.0
15	Burundi	bi	2010	6.493265915	2032135192	216.7277053	1.616	3.554153982	8.561555164	5.1
16	Belgium	be	2010	2.189299204	4.81557E+11	44197.39367	8.293		1.703104533	2.7
17	Benin	bj	2010	2.207835325	9535345011	973.2442545	1.04	4.195490191	0.88088567	2.1
18	Burkina Faso	bf	2010	-0.764230735	10109619741	624.9572522	3.723	1.285141169	3.779618728	8.4
19	Bangladesh	bd	2010	8.126676392	1.15276E+11	757.3852804	3.379	4.736123569	7.144663028	5.5
20	Bulgaria	bg	2010	2.438990605	50760929303	6863.667068	10.279	10.3015198	0.764129879	1.5
21	Bahrain	bh	2010	1.961884683	26805984043	21819.32911	1.13	-4.249245845	12.0069507	4.3
22	Bahamas, The	bs	2010	1.344027384	10095760000	27473.10043	14.432	5.915788916	-0.392565566	1.5

The arrow shows the different years represented on different sheets in the same work book

CHAPTER 3 ASSIGNMENT TWO

Question one of assignment two was a continuation of question one assignment two and it was about data visualization and analysis

We used the data set from question 1 of assignment 1 to plot different patterns of graphs

Due to the large size of datasets from kaggle we extracted parts of the dataset to plot graphs.

3.1 Commonly used commands

```
“Ylabel”……….labels the y axis
“xlabel”…………labels the x axis
“title “……………writes the title on the plot
“figure”………….. plots a singular figure without overwriting the previous
```

We started with a graph of unemployment rates in 2010

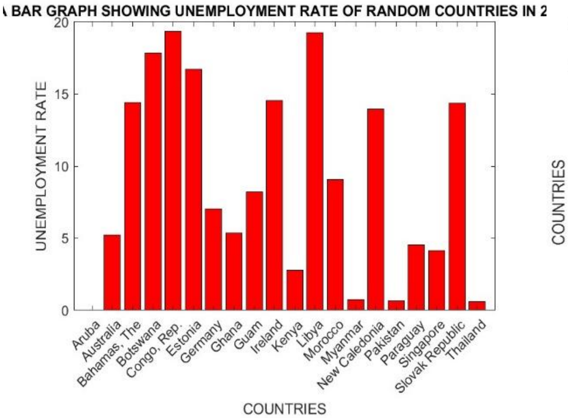
```
%UNEMPLOYMENT RATE 2010
figure;
Countrynames2010=MS2010.country_name(1:10:200);
Unemployment2010=MS2010.UnemploymentRate___(1:10:200);
bar(categorical(Countrynames2010),Unemployment2010,'r');
xlabel('COUNTRIES');
ylabel('UNEMPLOYMENT RATE');
title('A BAR GRAPH SHOWING UNEMPLOYMENT RATE OF RANDOM COUNTRIES IN 2010')
%UNEMPLOYMENT RATE 2011
figure;
Countrynames2011=MS2011.country_name(1:10:200);
Unemployment2011=MS2011.UnemploymentRate___(1:10:200);
barh(categorical(Countrynames2011),Unemployment2011,'g');
ylabel('COUNTRIES');
xlabel('UNEMPLOYMENT RATE');
title('A BAR GRAPH SHOWING UNEMPLOYMENT RATE OF RANDOM COUNTRIES IN 2011'),
%2012 inflation cpl against gdp current
```

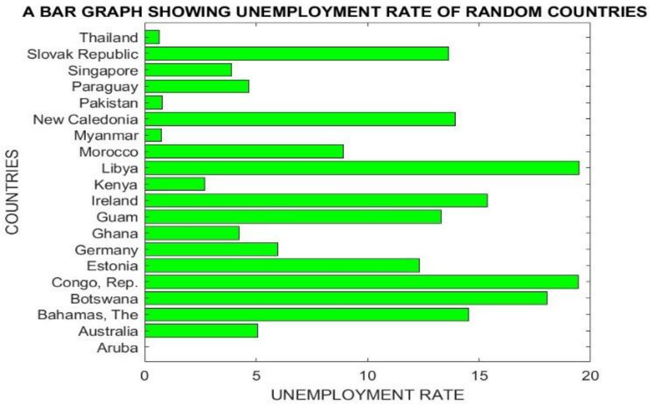
The code takes every 10th country (rows 1up to 200) to avoid overcrowding.

Plots unemployment rates for **2010** (red vertical bar) and **2011**(horizontal green bar graph)

categorical(): Converts country names to categorical data for proper bar chart formatting

3.2 Illustration





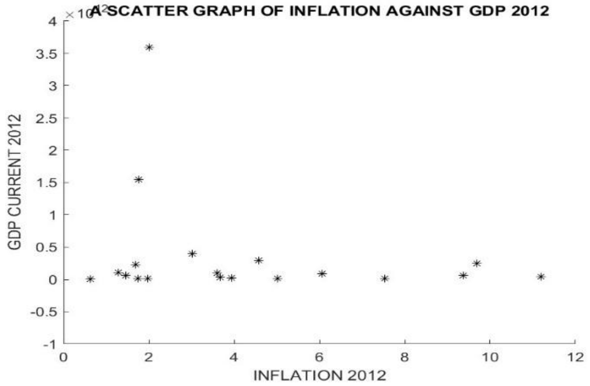
3.3 Scatter plot Inflation vs GDP

```
%2012 inflation cpl against gdp current
figure;
T=table(MS2012.country_name(1:10:200),MS2012.Inflation_CPI__(1:10:200),MS2012.GDP_CurrentUSD_(1:10:200),'VariableNames',{'countrynames','inflation','GDP_Current'});
```

The code creates a table having country names, inflation_CPI_and GDP CURRENT of year 2012,picking the 10th countries from 1 to 200

```
scatter(T.inflation,T.GDP_Current,'k*');
xlabel('INFLATION 2012');
ylabel('GDP CURRENT 2012');
title('A SCATTER GRAPH OF INFLATION AGAINST GDP 2012');
ylim([-1e12,4e12]);
 The code Examines correlation between inflation and GDP
'k*': Black asterisk markers
ylim(): Sets y-axis limits from -1 trillion to 4 trillion USD
```

3.2.1 ILLUSTRATION



3.3 %2013 government expense sample

```
 T2013=table(MS2013.country_name(17:24),MS2013.GovernmentExpense__OfGDP_(17:2
4),'VariableNames',{'countrynames','governmentexp'});
```

The code creates a table with country name and government expense

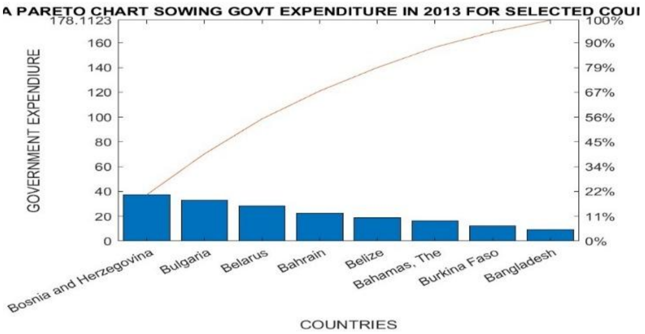
```
 T2013sorted=sortrows(T2013,'governmentexp','descend');
 figure;
 pareto(T2013sorted.governmentexp,T2013sorted.countrynames);
 xlabel('COUNTRIES');
 ylabel('GOVERNMENT EXPENDIURE');
 title('A PARETO CHART SOWING GOVT EXPENDITURE IN 2013 FOR SELECTED
COUNTRIES')
```

Pareto Principle: Shows "vital few" vs "trivial many"

Sorting: Countries sorted by expenditure (highest to lowest)

Output: Bar chart + cumulative percentage line

3.3.1 illustration



3.4 % %2014 GOVERNMENT REVENUE

```
 T2014=table(MS2014.country_name(1:5:200),MS2014.GovernmentRevenue__OfGDP_(1:
5:200),'VariableNames',{'countrynames','government_revenue'});
 figure;
 histogram(T2014.government_revenue);
```

```
ylabel('GOVERNMENT REVENUE');
```

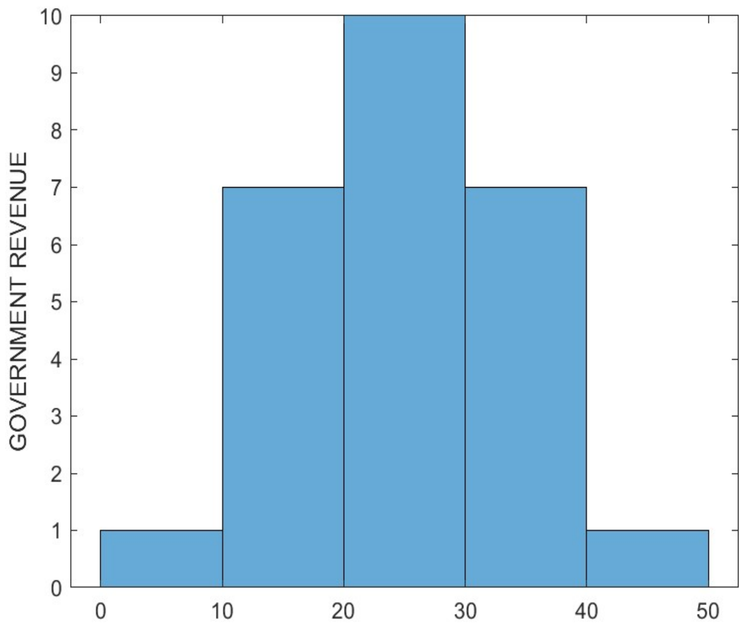
The table creates a table having country names and government_revenue

histogram(T2014.government_revenue);

Distribution: Shows frequency distribution of government revenue as % of GDP

Bins: Automatic binning of continuous data

3.4.1 Illustration

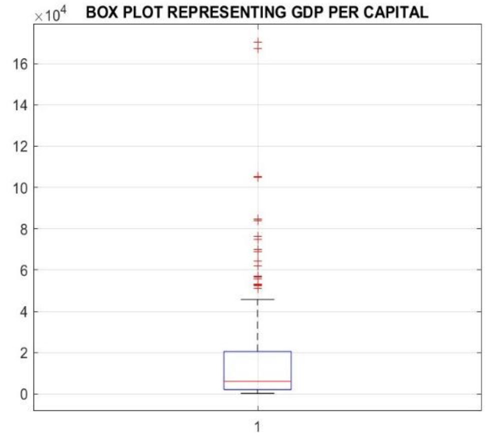


3.5 Box plot

```
%2015 GDP
figure;
boxplot(MS2015.GDPPerCapita_CurrentUSD_);
grid on;
title('BOX PLOT REPRESENTING GDP PER CAPITAL');
```

The code creats a box plot showing GDPPerCapita_CurrentUSD_) in the table ,MS2015

3.5.1 Illustration



3.6 PIE CHART

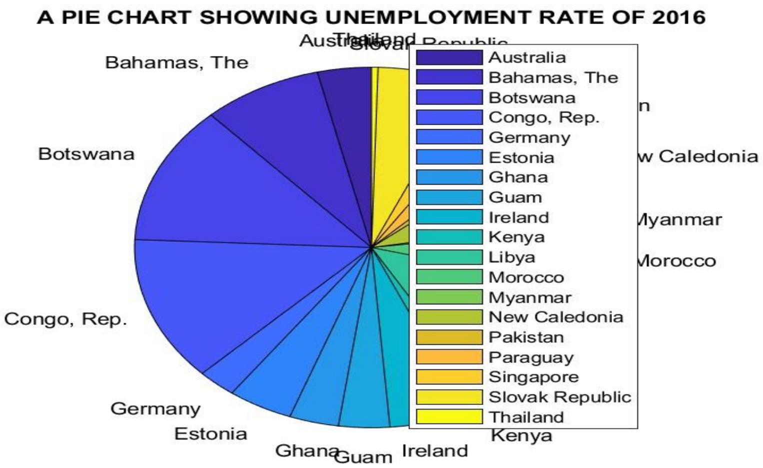
```
%2016 PIECHART
T2016=table(MS2016.country_name(1:10:200),MS2016.UnemploymentRate___(1:10:200),'VariableNames',{'countrynames','UnemploymentRate'});
Finite_idx=isfinite(T2016.UnemploymentRate);
fincountry=T2016.countrynames(Finite_idx);
finemp=T2016.UnemploymentRate(Finite_idx);
figure;
pie(finemp,fincountry);
title('A PIE CHART SHOWING UNEMPLOYMENT RATE OF 2016');
legend;
```

The code creates a table having country names and un employmnent rate

Data Cleaning: isfinite() removes NaN/Inf values

Proportional View: Each slice represents a country's unemployment rate proportion

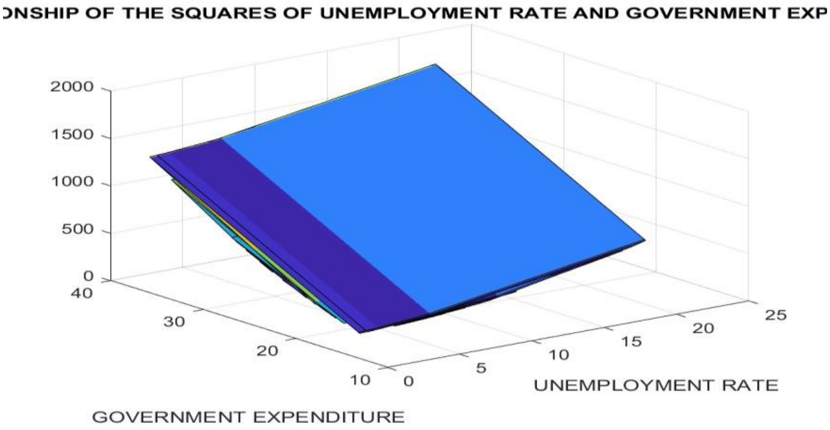
3.6.1 Illustration



3.7 MESH GRID

```
%2017 CATEGORIZATION
T2017=table(MS2017.country_name(1:10:200),MS2017.UnemploymentRate___(1:10:200),MS2017.GovernmentExpense__OfGDP_(1:10:200),'VariableNames',{'countrynames','UnemploymentRate','Governmentexp'});
[X, Y]=meshgrid(T2017.UnemploymentRate,T2017.Governmentexp);
figure;
Z=X.^2+Y.^2;
surf(X,Y,Z);
xlabel('UNEMPLOYMENT RATE');
ylabel('GOVERNMENT EXPENDITURE');
title('RELATIONSHIP OF THE SQUARES OF UNEMPLOYMENT RATE AND GOVERNMENT EXPENDITURE');
meshgrid(): Creates 2D grid from 1D vectors
Z=X.^2+Y.^2: Calculates squared distance function
surf(): 3D colored surface plot
```

3.7.1 ILLUSTRATION

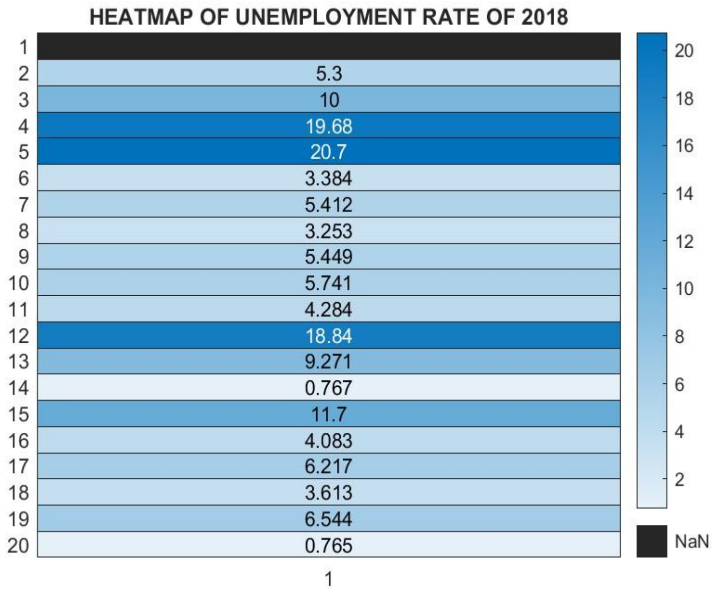


3.8 HEAT MAP

3.8.1%UNEMPLOYMENT RATE 2018

```
figure;
heatmap(MS2018.UnemploymentRate___(1:10:200));
title('HEATMAP OF UNEMPLOYMENT RATE OF 2018');
The code creates a heatmap using unemployment rate
```

3.8.2 Illustration



3.9 Mesh grid

```
%2017 CATEGORIZATION
T2017=table(MS2017.country_name(1:10:200),MS2017.UnemploymentRate___(1:10:20
0),MS2017.GovernmentExpense__OfGDP_(1:10:200),'VariableNames',{'countrynames'
,'UnemploymentRate','Governmentexp'});
[X, Y]=meshgrid(T2017.UnemploymentRate,T2017.Governmentexp);
figure;
Z=X.^2+Y.^2;
surfc(X,Y,Z);
xlabel('UNEMPLOYMENT RATE');
ylabel('GOVERNMENT EXPENDITURE');
title('RELATIONSHIP OF THE SQUARES OF UNEMPLOYMENT RATE AND GOVERNMENT
EXPENDITURE');
```

The code creates a table having country names, unemployment rate, government expense

Creates a new table T2017 with selected data

Sampling: 1:10:200 selects every 10th row from the 2017 data (approximately 20 countries)

Variables extracted:

Country names

Unemployment rates (%)

Government expenditure (% of GDP)

Result: A table with ~20 rows and 3 columns containing sampled country data.

reates a **paraboloid surface** (3D parabola)

X.^2: Element-wise squaring of unemployment values

Y.^2: Element-wise squaring of expenditure values

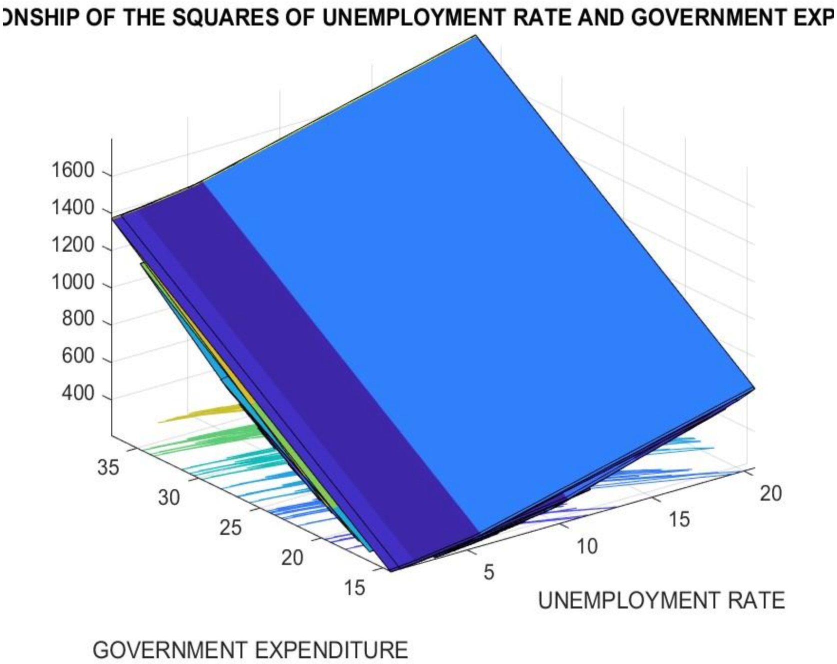
+: Sums the squared values

surf: Creates a 3D colored surface plot

c: Adds contour lines on the XY-plane beneath the surface

Combined: Surface + contour plot = "surfc"

3.9.1 Illustration



3.10 Quiver plot

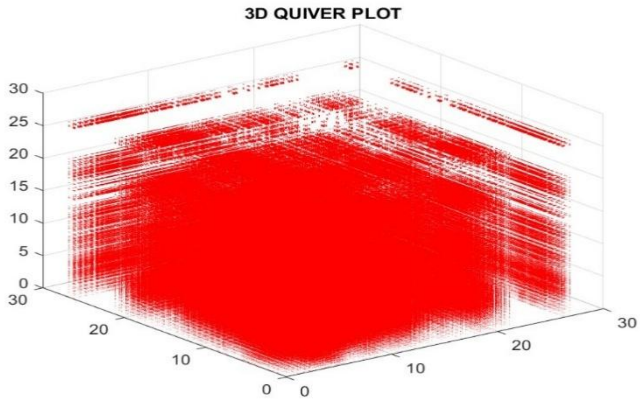
```
%2018 unemployment
figure;
M=MS2018.UnemploymentRate___;
[A B C]=meshgrid(M);
u=sin(A);
v=sin(B);
w=C;
quiver3(A,B,C,u,v,w,'r');
title('3D QUIVER PLOT');
```

Quiver plot: Shows vector fields based on unemployment data.

Vector Field: Shows direction and magnitude

Arrow Plot: Red arrows in 3D space

3.10.1 Illustration



3.11 3D STEM PLOT

```
figure;
F=MS2019.UnemploymentRate___;
y1=sin(F);
z1=cos(F);
stem3(F,y1,z1,'g');
```

title('3D STEM PLOT');

Stem plot: Plots unemployment rates (2019) with sine and cosine transformations.

Discrete 3D Data: Points with vertical lines to xy-plane

Trigonometric: Plots unemployment vs its sine/cosine

CHAPTER 4 QUESTION TWO

4.1Assignment 1(b): Capturing Members' Attributes in MATLAB

4.1.1 Question

Each group has different members from different backgrounds, home districts, courses, sex, villages, tribes, religions, regions, causes, interests, ages, names, and facial representations. Write a MATLAB code that captures each member's information/attributes into a single variable. Ensure the code uses the variable.

Requirements:
- Report
- PowerPoint presentation

Steps Taken

1. We launched MATLAB and opened a new script editor where the code was typed and saved.
2. A table structure was created in MATLAB with predefined variable names such as Name, Tribe,

Age, Interests, Village, District, Religion, Course, and Facial Representation.

4.1.2 MATLAB Code

```
 % defining struct
Members=struct('name',{},'age',{},'tribe',{},'interests',{},'village',{},'rel
igion',{},'course',{},'district',{},'facialrepresentation',{});
%Member 1
Member(1).name='SSEBAKIJJE HOSEA';
Member(1).age=21;
Member(1).tribe='MUGANDA';
Member(1).interests='SINGING';
Member(1).village='NAKISUNGA';
Member(1).religion='SDA';
Member(1).course='AMI';
Member(1).district='MUKONO';
Member(1).facialrepresentation='C:\Users\sseba\Desktop\assignment two\PICS
GROUP 7\DSC_6414.JPG';
%Member 2
Member(2).name = 'NYERO MARIA';
member(2).age=22;
Member(2).tribe = 'LANGI';
Member(2).interests = 'COOKING';
Member(2).village = 'NGETTA';
Member(2).religion = 'CATHOLIC';
Member(2).course = 'WAR';
Member(2).district = 'LIRA';
Member(2).facialrepresentation = 'C:\Users\sseba\Desktop\assignment two\PICS
GROUP 7\maria.jpg';
%member 3
Member(3).name = 'KAKURU OBADIAH';
Member(3).age = 22;
Member(3).tribe = 'MUNYANKOLE';
Member(3).interests = 'FOOTBALL';
Member(3).religion = 'ANGLICAN';
Member(3).course = 'WAR';
Member(3).district = 'NTUNGAMO';
Member(3).facialrepresentation = 'C:\Users\sseba\Desktop\assignment two\PICS
GROUP 7\OBADIA.jpg';
% Member4
Member(4).name = 'Sembera sherina Tapiness';
Member(4).age = 21;
Member(4).tribe = 'Musoga';
Member(4).interests = 'Eating';
Member(4).village = 'Bugenbe';
Member(4).religion = 'Anglican';
Member(4).course = 'meb';
Member(4).course = 'Meb';
```

```
Member(4).district = 'Jinja';
Member(4).facialrepresentation = 'C:\Users\sseba\Desktop\assignment two\PICS
GROUP 7\SHERINA.jpg';
%Member 5
Member(5).name = 'Esarait Brian';
Member(5).age = 21;
Member(5).tribe = 'Iteso';
Member(5).interests = 'RUGY';
Member(5).village = 'SOROTI';
Member(5).religion = 'ANGLICAN';
Member(5).course = 'MEB';
Member(5).district = 'SOROTI';
Member(5).facialrepresentation = 'C:\Users\sseba\Desktop\assignment two\PICS
GROUP 7\ESIRIAT.jpg';
% Member 6
Member(6).name = 'Adipo Hope Odware';
Member(6).age = 21;
Member(6).tribe = 'Atesot';
Member(6).interests = 'Reading';
Member(6).village = 'Bukedea';
Member(6).religion = 'BORNAGAIN';
Member(6).course = 'MEB';
Member(6).district = 'BUKEDEA';
% Member 7
Member(7).name = 'GUDOI ALLAN';
Member(7).age = 22;
Member(7).tribe = 'Mugisu';
Member(7).interests = 'Research';
Member(7).village = 'Majanga';
Member(7).religion = 'Anglican';
Member(7).course = 'WAR';
Member(7).district = 'Mbale';
Member(7).facialrepresentation = 'C:\Users\sseba\Desktop\assignment two\PICS
GROUP 7\GUDOI.jpg';
% Member 8
Member(8).name = 'Kobusingye Bethline';
Member(8).course = 'PTI';
Member(8).district = 'Ntungamo';
Member(8).village = 'Itojo';
Member(8).tribe = 'Munyankore';
Member(8).religion = 'Protestant';
Member(8).age = 21 ;
Member(8).interests = 'Watching movies';
Member(8).facialrepresentation = 'C:\Users\sseba\Desktop\assignment two\PICS
GROUP 7\bethelene.jpg';
%Member 9
Member(9).name = 'ETYANG JONATHAN';
Member(9).age = 22;
Member(9).interests = 'MATLAB';
```

```
Member(9).district = 'JINJA';
Member(9).course = 'APE';
```

Example Run in MATLAB

Below is an example screenshot of how the code runs in MATLAB:


Variables - Member
Member
1x9 struct with 9 fields

Fields	name	age	tribe	interests	village	religion	course	district	facialrepresentation
1	'SSEBAKIJJ...	20	'MUGANDA'	'SINGING'	'NAKISUN...	'SDA'	'AMI'	'MUKONO'	1800x1200x3 uint8
2	'NYERO M...	[]	'LANGI'	'COOKING'	'NGETTA'	'CATHOLIC'	'WAR'	'LIRA'	1187x1149x3 uint8
3	'KAKURU O...	22	'MUNYANK...	'FOOTBALL'	[]	'ANGLICAN'	'WAR'	'NTUNGA...	1080x810x3 uint8
4	'Sembera s...	21	'Musoga'	'Eating'	'Bugenbe'	'Anglican'	'Meb'	'Jinja'	1280x854x3 uint8
5	'Esarait Bria...	21	'Iteso'	'RUGY'	'SOROTI'	'ANGLICAN'	'MEB'	'SOROTI'	1080x810x3 uint8
6	'Adipo Ho...	21	'Atesot'	'Reading'	'Bukedea'	'BORNAGA...	'MEB'	'BUKEDEA'	[]
7	'GUDOI AL...	22	'Mugisu'	'Research'	'Majanga'	'Anglican'	'WAR'	'Mbale'	1080x810x3 uint8
8	'Kobusingy...	21	'Munyanko...	'Watching ...	'Itojo'	'Protestant'	'PTI'	'Ntungamo'	1080x703x3 uint8
9	'ETYANG J...	22	[]	'MATLAB'	[]	[]	'APE'	'JINJA'	[]

Editor - C:\Users\sseba\Desktop\assignment two\numbertwo.m
Workspace

Figure 1: Table of members created after running the MATLAB code.

4.1.3 Explanation of the Code

The code defines a structure array called Members with the listed fields (name, age, tribe, etc.), but initializes it as empty ({} means no initial data).

We now have a template structure where each future element (e.g., Members(1), Members(2)) will contain the same set of fields.

The code Fills in each field of the **first member** with their personal information but the facialrepresentation field stores a **file path** to an image file.

The same is done for all the members

Each **index** in Member represents **one person**.

Each **field** holds a specific type of information about that person.

Conclusion

This task demonstrated how MATLAB can be used for organizing qualitative and quantitative data. , tables, and structured data entry.

4.1.4 Assignment 2(b): Visualization of Members' Attributes

Question

Write a MATLAB code that captures each member's information/attributes into a single variable. Then, generate different statistical plots using the captured data such as bar graphs, scatter plots, box plots, pie charts, and other relevant visualizations.

4.1.4.1 Steps Taken

1. We extended the previous code to include visualization functions.
2. The program still stores all members' information in a structured variable.
3. MATLAB plotting functions were used to create bar graphs, scatter plots, box plots, pie charts, and more advanced visualizations such as mesh and geoplot.

4. Each figure represents a relationship between members' attributes such as age, names, or interests.

4.1.4.2 MATLAB Code

```
 %% Assignment 2
figure("Name","RELATION OF AGE AND NAMES")
age = [20 22 22 21 21 21 22 21 22];
Names = categorical({'SSEBAKIJJE HOSEA','NYERO MARIA','KAKURU
OBADIAH','Sembera sherina Tapiness','Esarait Brian','Adipo Hope
Odware','GUDOI ALLAN','Kobusingye Bethline','ETYANG JONATHAN'});
bar(Names,age)
xlabel('names');
ylabel('age');
title('RELATION OF AGE AND NAMES');
grid on
%stair plot of cos(age) aganist ages
figure("Name","stair plot of cos(age)aganist ages" )
c = cos(age);
s = sin(age);
stairs(age,c)
xlabel('age');
ylabel('c');
grid on;
title('cos(age) aganist age');
% horizontal bar graph
figure("Name","HORIZONTAL BAR GRAPH)")
barh(Names,age);
xlabel('AGE');
ylabel('NAME');
title('HORIZONTAL GRAPH')
grid on
%scatter graph of age and sin(age)
figure("Name","scatter graph");
scatter(age,s);
xlabel('AGE');
ylabel('sin(age)')
title('scatter graph')
grid on
%line plot
figure('Name','line plot');
plot(c,s,'r')
xlabel('cos(age)');
ylabel('sin(age)');
title('line plot')
grid on
%stem plot
figure('Name','stem plot')
stem(c,s)
```

```
xlabel('cos(age)');
ylabel('sin(age)');
title('stem plot of sin(age)and cos(age)')
grid on
%step plot
figure('Name','step plot')
step(c,age)
xlabel('cos(age)');
ylabel('age');
title('step plot of age and cos(age)');
grid on
%error bar plot
figure('Name','error bar plot')
errors = age-4;
errorbar(c,age,errors)
xlabel('cos(age)');
ylabel('age');
title('error bar plot');
grid on
%Area plot
figure('Name','Area plot');
area(errors,age);
xlabel('errors');
ylabel('age');
title('Area plot');
grid on
%Histogram
figure('Name','Histogarm');
histogram(age);
xlabel('AGE');
title('HISTOGRAM OF AGE');
grid on;
%pareto chert
figure('Name','PARETO CHART');
pareto(age);
title('PARETO CHART');
%BOX PLOT
figure('Name','Box plot');
boxplot(age);
title('BOX PLOT');
%pie chart
figure('Name','PIECHART OF AGES');
pie(age);
title('PIECHART OF AGES');
%LOGARITHIMIC PLOT
figure('Name','LOG CURVE');
D = exp(s);
semilogx(age,D);
title('LOG CURVE OF AGE');
```

```
grid on
%stem 3 plot
figure('Name','STEM 3 PLOT');
stem3(age,s,c);
title('3D STEM PLOT');
%SURFACE PLOT
figure('Name','SURFACE PLOT');
x = 20:0.5:22
[x,y] = meshgrid(x);
z = x.*2+y.*4;
surf(x,y,z);
xlabel('x');
ylabel('y');
zlabel('z');
title('SURFACE PLOT');
%MESH PLOT
figure('Name','meshplot');
h = sin(x)+cos(y);
mesh(x,y,h);
xlabel('x');
ylabel('y');
zlabel('z');
title('MESH PLOT');
%WATER FALL PLOT
figure('Name','WATER FALL PLOT');
waterfall(x,y,z);
xlabel('xvalues');
ylabel('yvalues');
zlabel('zvalues');
title('water fall plot')
%SCATTER PLOT
figure('Name','SCATTER PLOT');
scatter3(x,y,z);
xlabel('xvalues');
ylabel('yvalues');
zlabel('zvalues');
title('SCATTER PLOT')
%3D BAR
figure('Name','BAR')
bar3(age)
title('D BARGRAPH')
%GEOPLOT
figure('Name','GEOPLOT');
geoplot(age,s)
title('GEOPLOT')
```

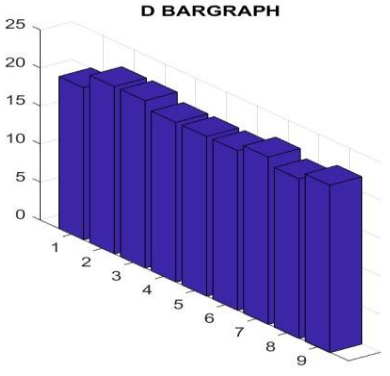
4.2 Common codes used

```
“Ylabel”……….labels the y axis
“xlabel”…………labels the x axis
```

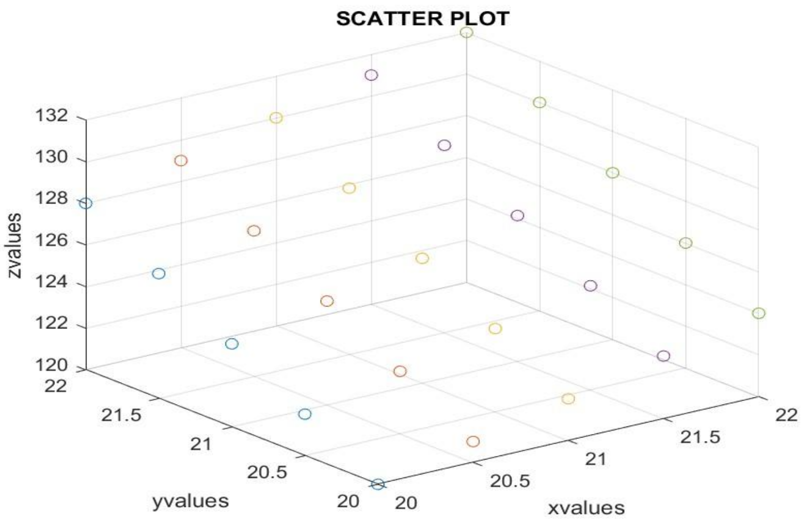
```
"title "…………….writes the title on the plot
"figure"…………… plots a singular figure without overwriting the previous
```

4.2.1 Generated Plots

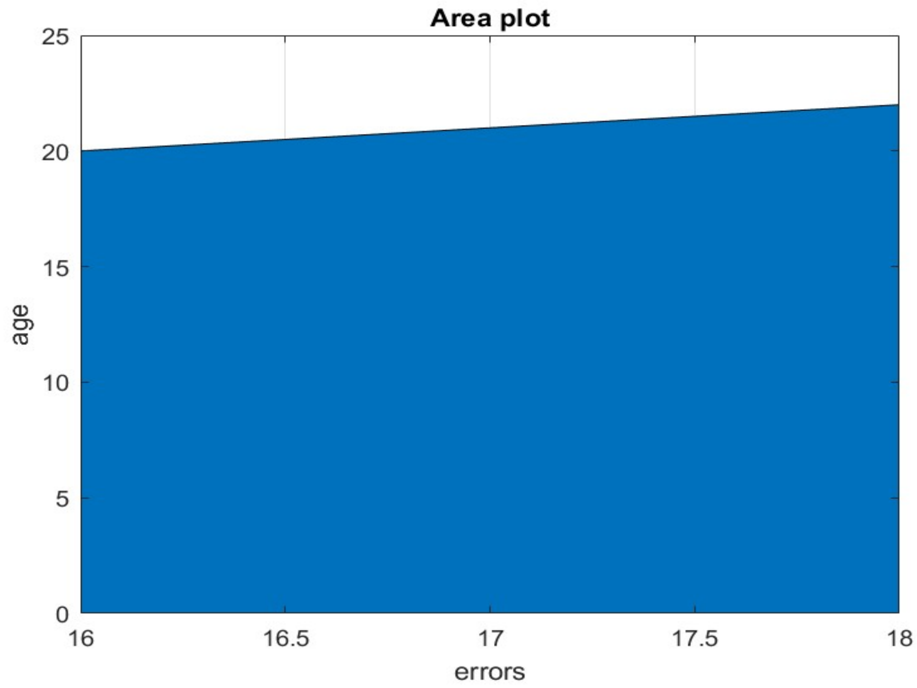
3D BAR GRAPH



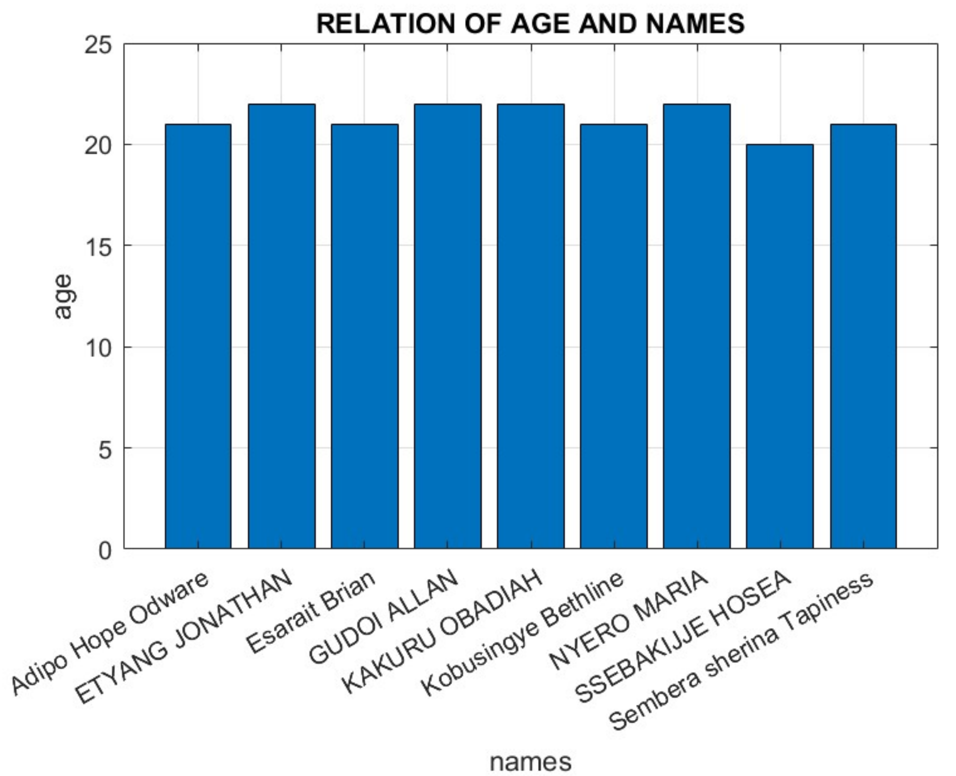
3D Scatter Plot of Age-related Values



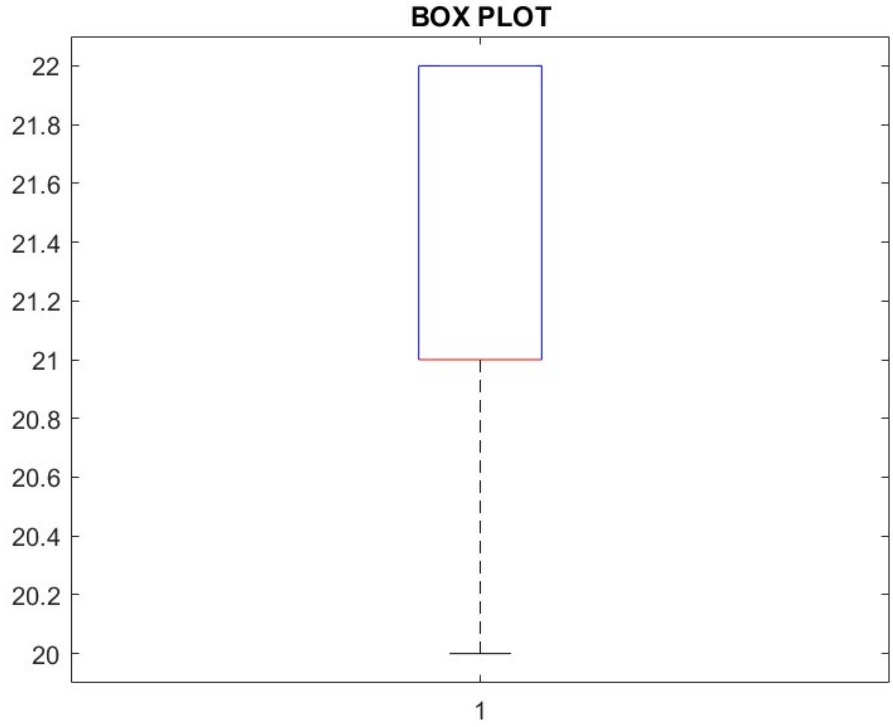
Area Plot of Age Distribution



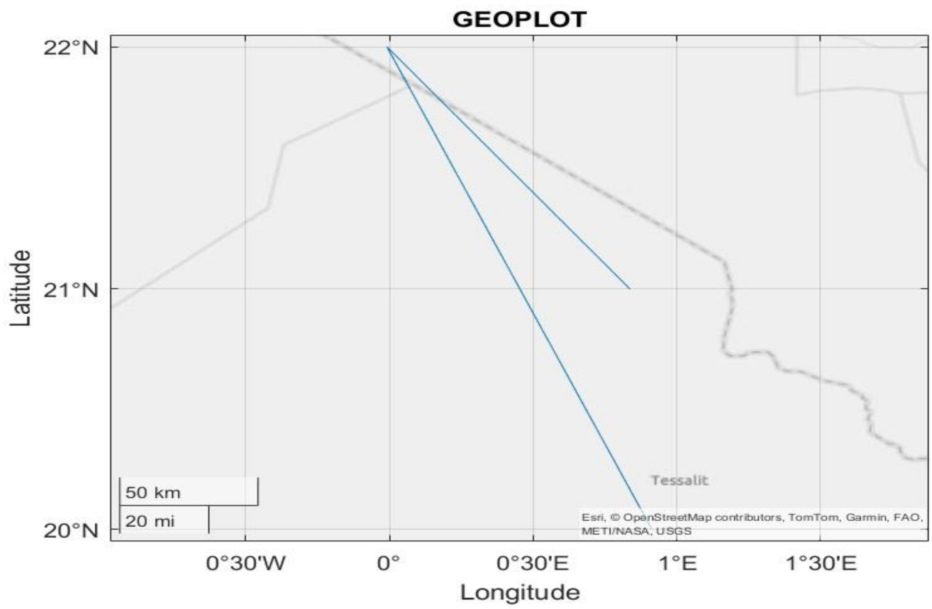
Bar Graph Showing Relation of Age and Names



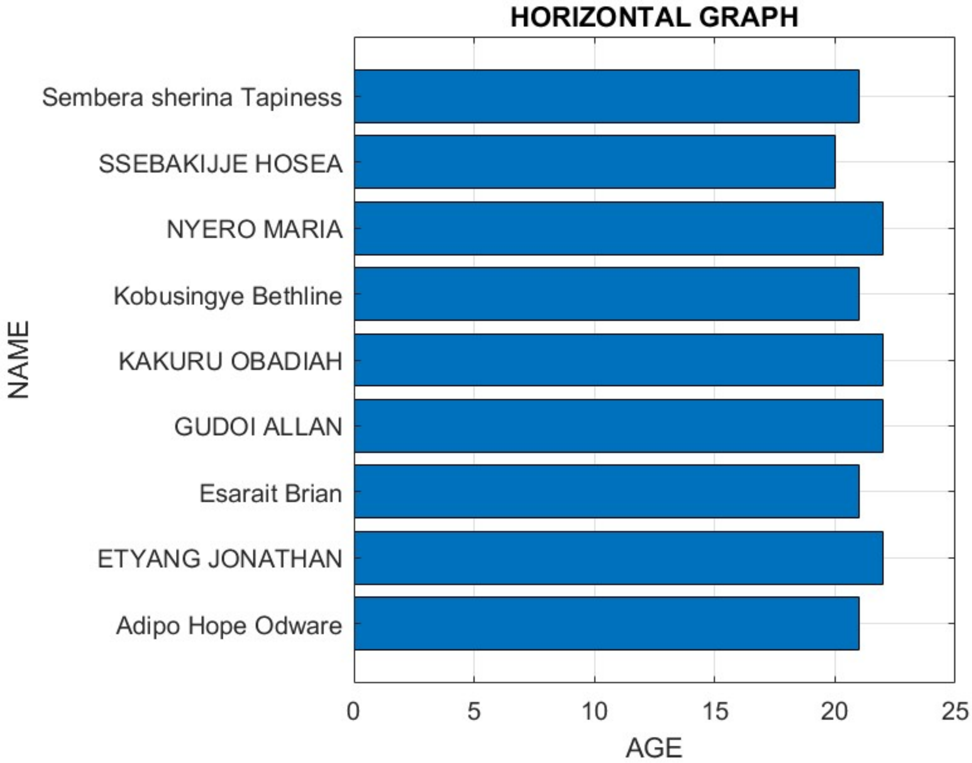
Box Plot Representing Age Spread



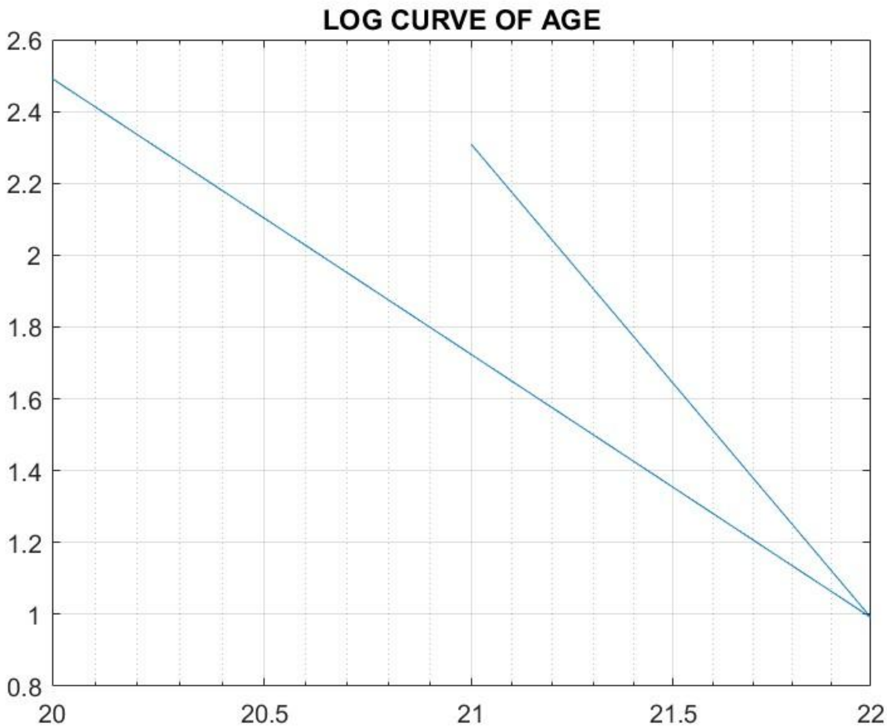
Geoplot Representation of Member Data



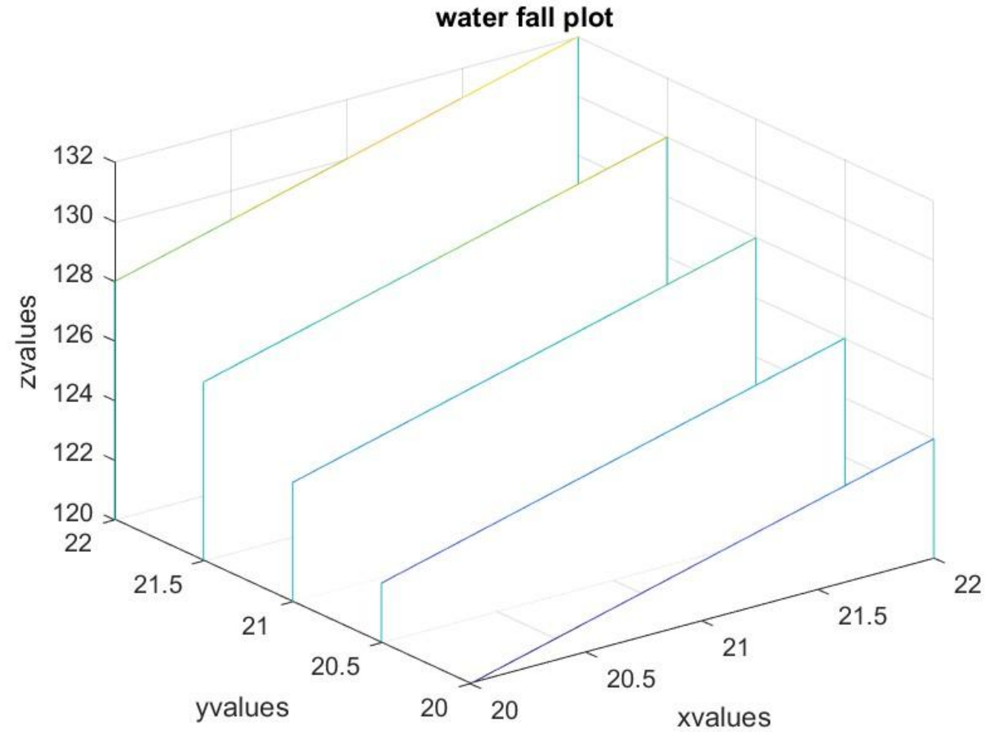
Horizontal Bar Graph of Names and Ages



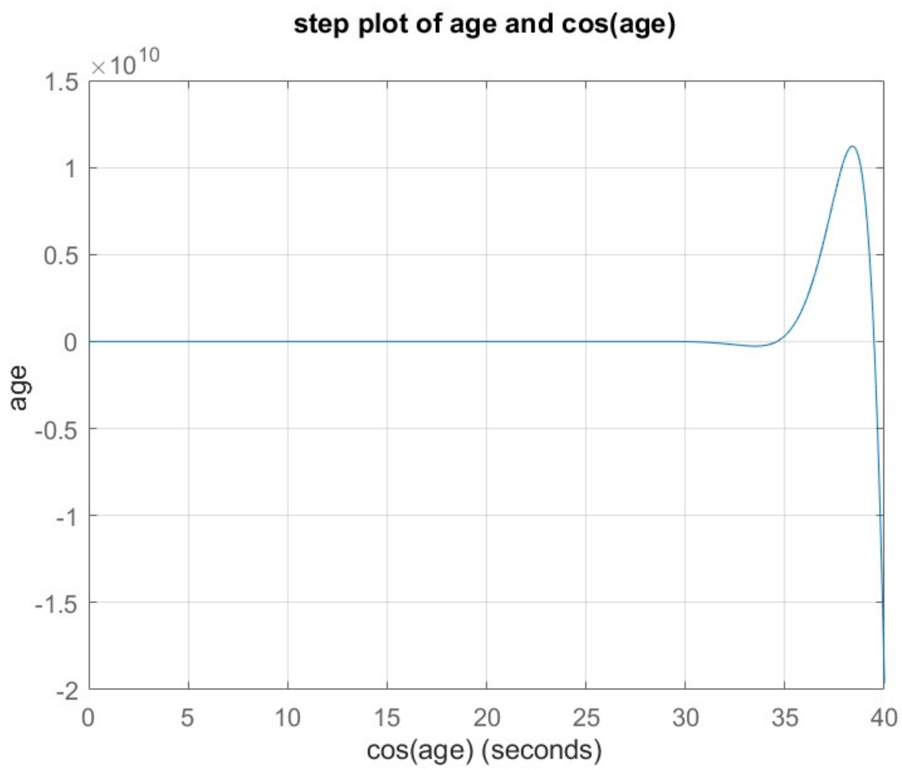
Logarithmic Curve of Age



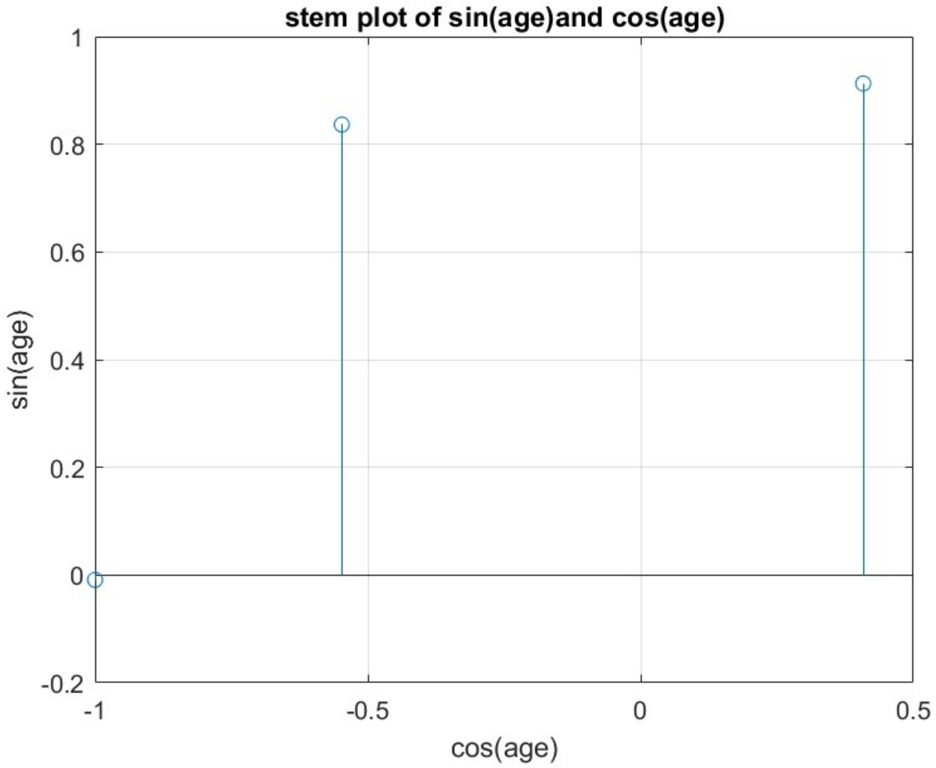
Waterfall Plot of Age Data



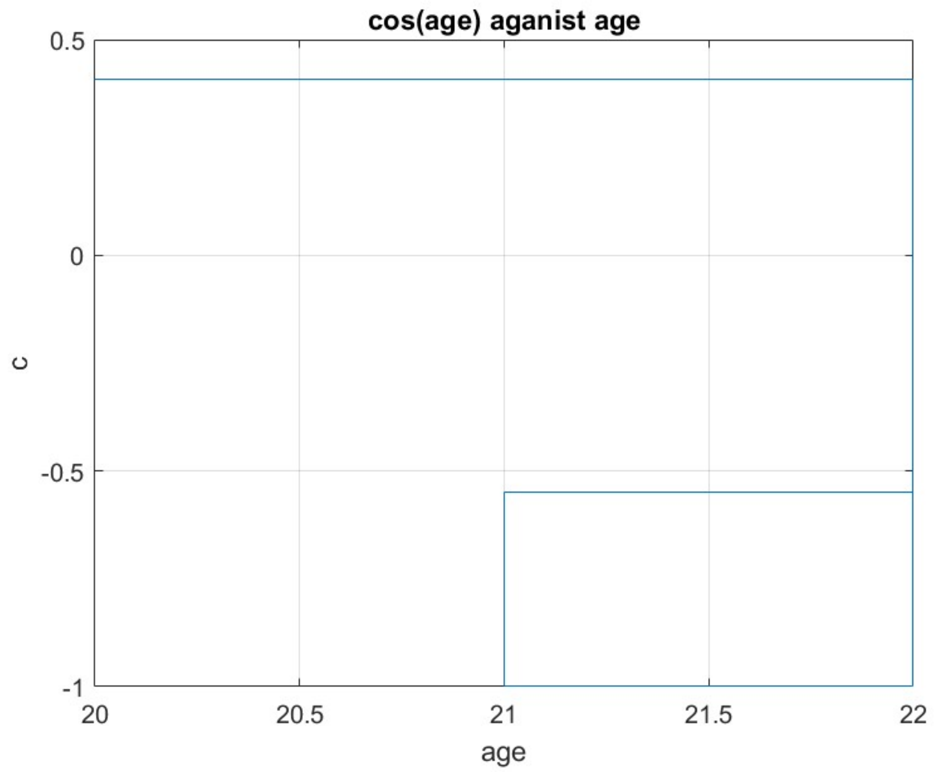
Step Plot of Age vs cos(Age)



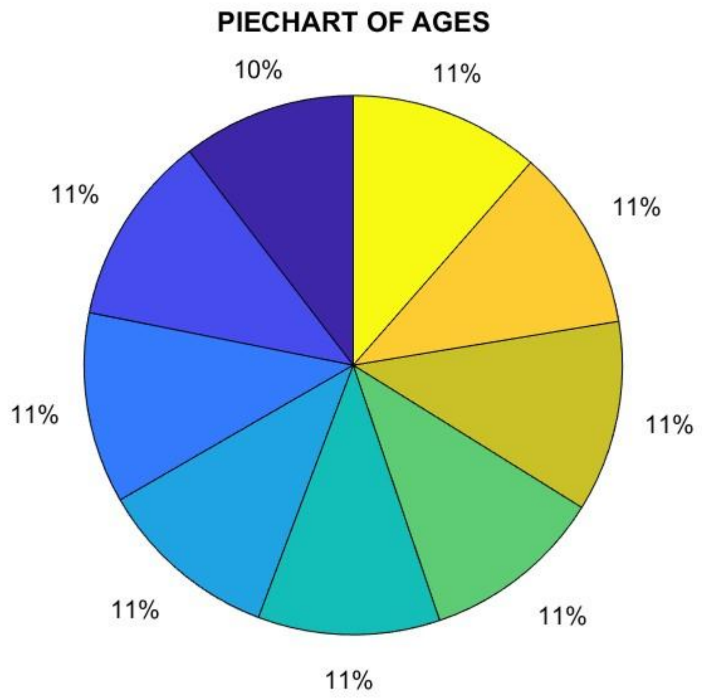
Stem Plot of sin(Age) vs cos(Age)



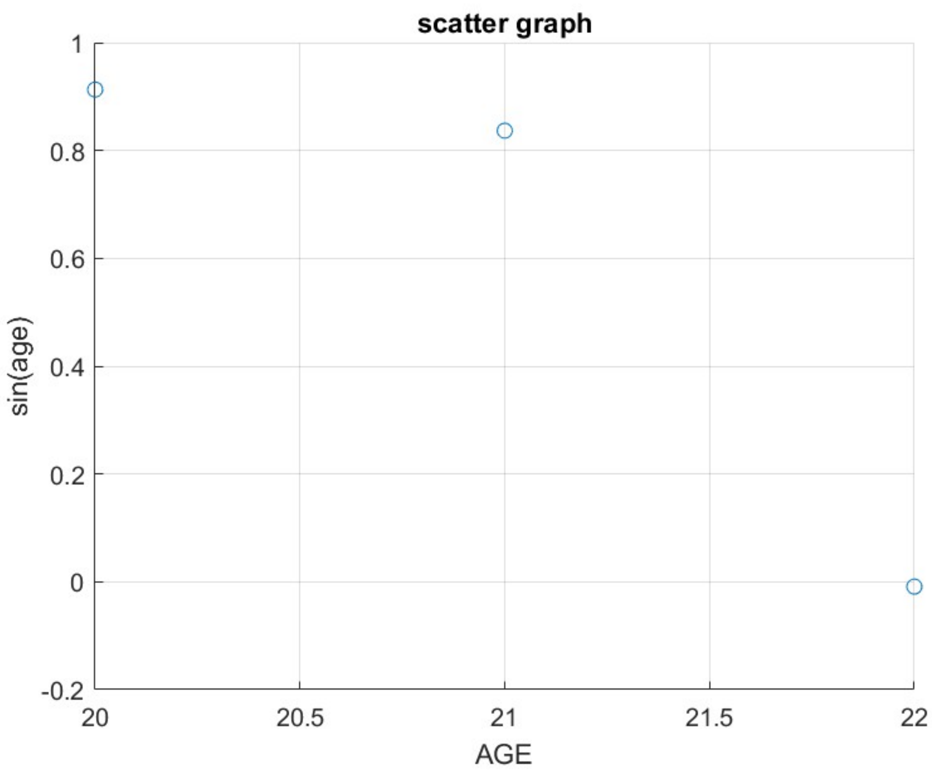
Stair Plot of cos(Age) against Age



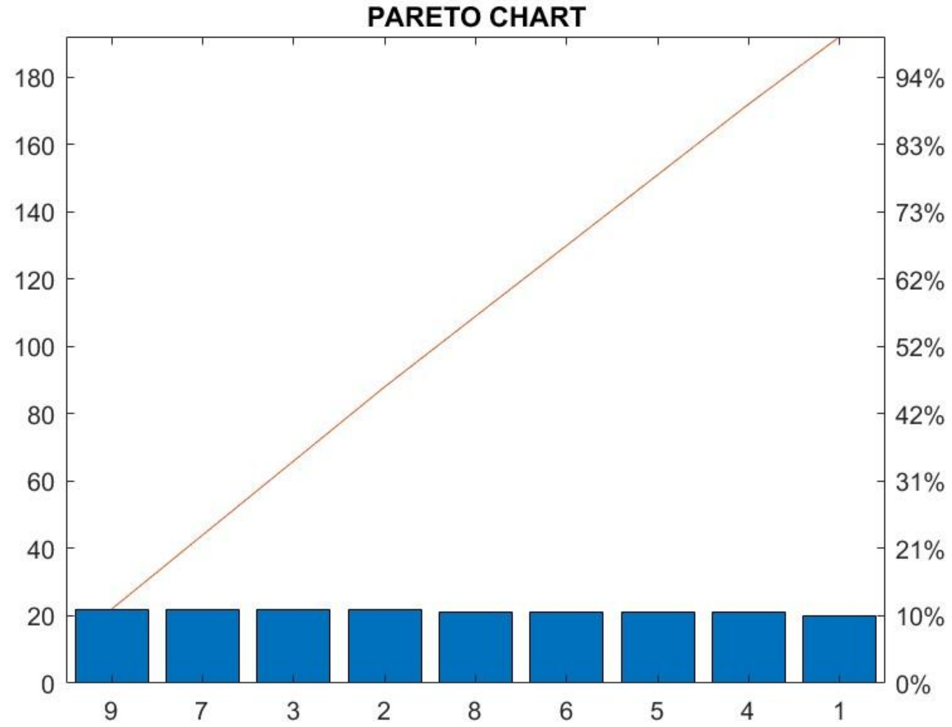
Pie Chart of Ages



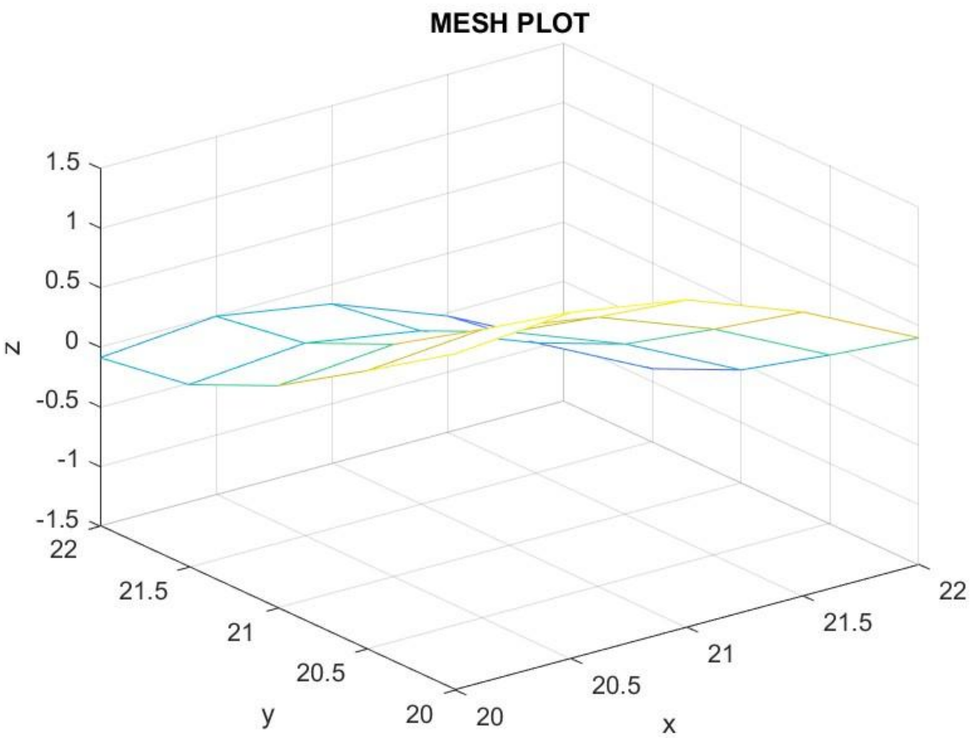
Scatter Graph of Age vs sin(Age)



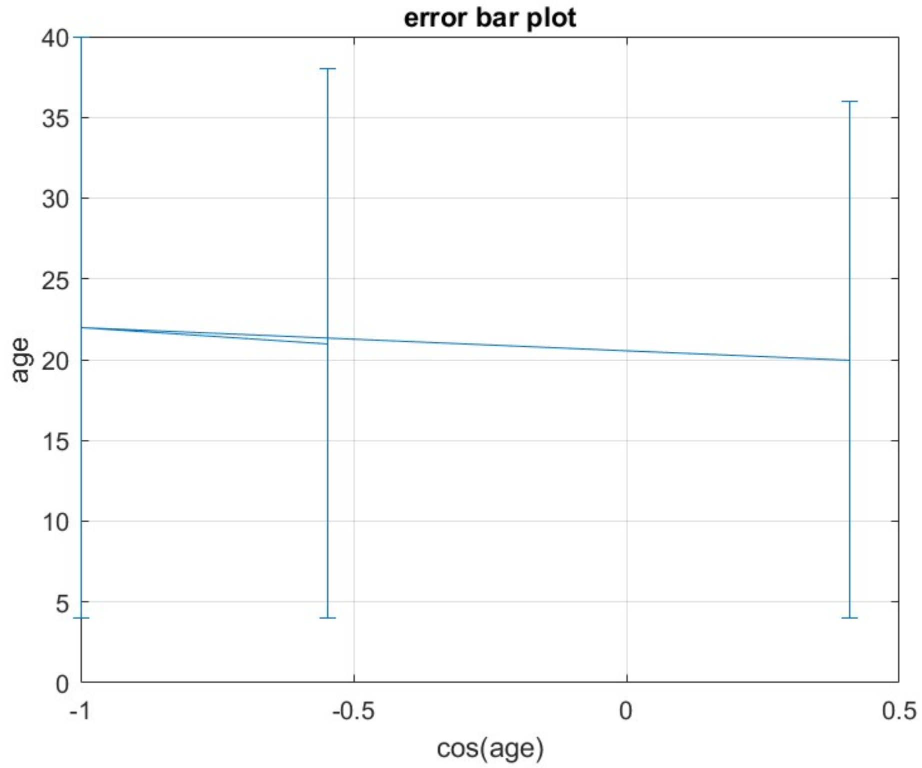
Pareto Chart of Age Data



Mesh Plot of Age Data



Error Bar Plot of Age Data



Explanation of the Code and Plots

The code first captures the attributes of all group members in a struct variable. It then applies various plotting functions available in MATLAB to visualize the data. Each plot gives insights into different aspects:

- Bar and horizontal bar graphs show the relationship between names and ages.
- Scatter and 3D scatter plots reveal point distributions in 2D and 3D.
- Box plot shows spread and outliers in age data.
- Pie chart shows proportional distribution of ages.
- Area, step, stem, stair, and waterfall plots provide different mathematical views of age values.
- Mesh plot visualizes data surfaces.
- Geoplot places members on a geographical context (if coordinates exist).
- Pareto and error bar charts provide advanced statistical insights.

Conclusion

This assignment extended our work by applying MATLAB's visualization capabilities to real data. We learned how to use multiple plotting functions to highlight relationships and patterns in data. It demonstrated the usefulness of MATLAB in both data management and advanced statistical visualization.