Application of Machine Learning In Economics

Project report submitted in partial fulfillment
of the requirements for the degree of

Bachelor of Technology
in
Engineering

by

Shashank Shankar - 18ucs135
Kumar Raghav Sharma - 18ucc156
Yash Baheti - 18ucs226

Under Guidance of
Dr.Surinder Singh Nehra

Department of Humanities and Sciences
The LNM Institute of Information Technology, Jaipur

May 2021

The LNM Institute of Information Technology
Jaipur, India

CERTIFICATE

This is to certify that the project entitled "Application of Machine Learning in Economics" , submitted by Shashank Shankar (18ucs135), Kumar Raghav Sharma (18ucc156) and Yash Baheti (18ucs226) in partial fulfillment of the requirement of degree in Bachelor of Technology (B. Tech), is a bonafide record of work carried out by them at the Department of Humanities and Social Sciences, The LNM Institute of Information Technology, Jaipur, (Rajasthan) India, during the academic session 2021-2022 under my supervision and guidance and the same has not been submitted elsewhere for award of any other degree. In my/our opinion, this thesis is of standard required for the award of the degree of Bachelor of Technology

21-01-2022

(B. Tech).

DateAdviser: Dr Surinder Singh Nehra.

Acknowledgments

We would like to express our heartfelt gratitude to Our Supervisor, Dr. Surendra Singh Nehra, for providing us with the wonderful opportunity to work on this wonderful project on the topic of Application of machine learning in economics. He also assisted us in conducting extensive research and introducing us to many new things, for which we are extremely grateful. Second, we'd want to express our gratitude to all of our friends who assisted us in completing this project within the time constraints.

Abstract

The validity of utilising machine learning to forecast economic variables is investigated in this research. Artificial intelligence (AI) has developed and is fast integrating into our society, and the economic prediction is no exception. Although AI has been utilised in the real world for economic forecasting, few research have focused on machine learning. This research focuses on machine learning, but it also compares it to a standard statistical model, the autoregressive (AR) model. Employing current data from G7 countries, a comparison of using an AR model and machine learning (LSTM) to anticipate GDP and consumer price is made. The empirical results demonstrate that the traditional forecasting AR model is slightly more accurate than the machine learning model, but there is little difference in consumer price predicting between the two.

Contents

Chapter Page

1 Introduction 1
1.1 Introduction to GDP 1
1.2 GDP sectors: Manufacturing, Service And Trading 1
1.2.1 Manufacturing 1
1.2.1.1 Market Size 1
1.2.1.2 Investments 2
1.2.1.3 Government Initiatives 2
1.2.1.4 Road Ahead 3
1.3 Service sector 4
1.4 Trading 5
1.4.1 Types of International Trade 5
1.4.2 Agriculture 6
1.4.2.1 Sectors in Agriculture 6
1.4.3 MSMEs 7
1.4.3.1 New Definition for MSMEs 7
1.4.3.2 Old Definition of MSMEs 7
1.4.3.3 Benefits of MSMEs 7
1.4.4 Exports,Market size and GDP: 7

2 Literature Survey 9

3 Proposed Work 10
3.1 Economic Forecasting and Machine Learning 10
3.2 Need for Macroeconomic Forecasting 10
3.3 Issues faced by Traditional Systems 11
3.4 Traditional Systems for Forecasting 12
3.4.1 Auto Regressive Moving Average Models ARIMA 12
3.4.2 Dynamic Factor Models 13
3.5 Data Rich Vs Data Poor Conditions 13

4 Simulation and Results 14
4.1 Libraries Used in Codes 14
4.2 Dataset 15
4.3 Code 1 15

4.3.1 Table Description 15
4.3.2 country Name 16
4.3.3 Description for India and USA 17
4.3.4 Splitting into Training and Test set 19
4.3.5 Classification 19
4.4 Code 2 23
4.4.1 Table Description 23
4.4.2 Region with Codes 23
4.4.3 GDP Plot 24
4.4.4 Correlation 25
4.4.5 Splitting into Training and Test set 27
4.4.6 Classification 28

5 Conclusions and Future Work 32
5.1 Scope of further work 32

Chapter *1*

Introduction

1.1 Introduction to GDP

The value of all final goods and services produced inside a country or an area over a period of time (a quarter or a year) is referred to as the gross domestic product (GDP), and it is frequently considered the best measure for gauging national economic circumstances (Mankiw & Taylor 2007). The Gross Domestic Product per capita (GDP per capita) is one of the most often used indicators of economic progress and one of the most important macroeconomic indicators. GDP per capita is a valuable metric for analysing a country's or region's macroeconomic position. Three factors contribute to the significance of GDP per capita as a measure of economic development:

- Degree of economic development
- Measure Social Justice and Equality
- Social Stability in the Country

1.2 GDP sectors: Manufacturing, Service And Trading

1.2.1 Manufacturing

Manufacturing has emerged as one of India's fastest-growing industries. Mr. Narendra Modi, India's Prime Minister, established the Make in India campaign to put India on the map as a manufacturing centre and to give the Indian economy international prominence. By 2022, the government wants to create 100 million new employment in the sector.

1.2.1.1 Market Size

According to the first advanced estimate for FY21, the sector's Gross Value Added (GVA) at current prices was anticipated to be US 350.27 billion. From 56.4 in December 2020, the IHS Markit India Manufacturing Purchasing Managers Index (PMI) climbed to 57.7 in January 2021. Capacity utilisation

in India's manufacturing sector was 63.3 percent in the second quarter of FY21, according to the latest study. India's industrial production, as measured by the Index of Industrial Production (IIP), was 135.9 in December 2020, according to the Ministry of Statistics Programme Implementation. Between April and December 2020, total merchandise exports totaled US dollar 200.80 billion.

1.2.1.2 Investments

With the help of the Make in India initiative, India is on its way to becoming a hi-tech manufacturing hub, as global giants such as GE, Siemens, HTC, Toshiba, and Boeing have either established or are planning to establish manufacturing plants in the country, attracted by India's market of over a billion consumers and rising purchasing power. Between April 2000 and September 2020, cumulative Foreign Direct Investment (FDI) in India's manufacturing sector totaled US$ 91.28 $billion. The Indian government boosted FDI in defence production through the automatic route from 49 percent to 74 percent in May 2020.. India has emerged as one of the most attractive manufacturing investment destinations. The following are some of the most significant investments and innovations in this area in recent years:

Amazon India announced on February 16, 2021 that it would begin producing electronic devices in India, beginning with the Amazon Fire TV stick. By the end of 2021, the business expects to begin manufacturing in Chennai with contract manufacturer Cloud Network Technology, a Foxconn affiliate. Toyota Kirloskar Motor (TKM) and the Directorate General of Training (DGT), Ministry of Skill Development and Entrepreneurship, inked a Memorandum of Understanding (MoU) in January 2021 to improve skills among the youth under the government's Flexi-MoU Scheme. Amazon announced on January 19, 2021 that it has formed a partnership with Startup India, Sequoia Capital India, and Fireside Ventures to launch an accelerator programme to help entrepreneurs deliver goods to global audiences. India's exports are quickly improving, and Amazon's Worldwide Selling programme is assisting in the development of Indian global brands. In 2019, e-commerce export sales (under the scheme) for more than 800 Indian MSMEs were USD 131,375 (Rs 1 crore).

1.2.1.3 Government Initiatives

The Indian government has taken a number of steps to foster a healthy environment for the country's industrial industry to thrive. The following are some important projects and developments: India's domestic growth in manufacturing, trade, and other sectors is predicted to be boosted by the Union Budget 2021-22. A main focal field is the development of a robust infrastructural, logistical, and utilities environment for the manufacturing industry. The following are some of these initiatives: Global industry champions will be established as a result of the Mega Investment Textiles Parks (MITRA) initiative to build world-class infrastructure, which will benefit from economies of scale and agglomeration. Over the course of three years, seven Textile Parks will be built. The government promised to invest heavily in the construction of modern fishing harbours and fish landing centres in Kochi, Chennai, Visakhapatnam, and other key fishing ports. In Tamil Nadu, there are two seaweed parks: Paradip and Petuaghat,

as well as a multipurpose Seaweed Park. Textile and marine exports are projected to benefit from these initiatives. The Ministry of the Food Processing Industry's 'Operation Green' scheme, which was previously limited to onions, potatoes, and tomatoes, has now been expanded to include 22 perishable products in order to boost agricultural exports. Infrastructure developments for horticultural products will be made easier as a result of this.

1.2.1.4 Road Ahead

India is a desirable location for manufacturing investments from throughout the world. Several mobile phone, luxury, and automobile brands, among others, have established or are considering doing so in the country. India's manufacturing sector has the potential to exceed US$1 trillion in revenue by 2025. India will become a common market with a GDP of US$ 2.5 trillion and a population of 1.32 billion people after the Goods and Services Tax (GST) is implemented, which would be a huge draw for investors. According to the Indian Cellular and Electronics Association (ICEA), legislative interventions might help India increase its total laptop and tablet production capacity to US$ 100 billion by 2025. To provide a holistic strategy, the government is focusing on the establishment of industrial corridors and smart cities. the nation's development The corridors will also help in integrating, monitoring, and building a favourable environment for industrial development, as well as promoting advanced manufacturing methods.

1.3 Service sector

The service sector has not only made waves in the internal economy, but it has also made an impact on the country's exterior commerce. India has risen to the top of the list in terms of service exports and imports. By 2006-07, India's trade-to-GDP ratio had risen from 22.50 percent in 2000-01 to 34.80 percent. However, when services trade is included, the increase is significantly greater, reaching 48 percent of GDP in 2006-07, up from 29.20 percent in 2000-01. (GOI 2008). In 2001, India's proportion of services trade as a percentage of total commerce was around 30%, significantly higher than the global average (WTO 2001). In compared to a meagre 5.40 percent annual average growth in merchandise exports, services exports have grown at a phenomenal rate of 23 percent. India's comparative advantage in service exports is not only bigger than its comparative advantage in products, but it is also growing (World Bank 2004). India's service exports increased by 19.34 percent over this time period, while imports increased by 15.91 percent. Globally, growth was 8.27 percent and 7.74 percent, respectively. The article addresses some of the major worries about India's economy's expansion, which has been fueled by its service sector..

In this article we will discuss about:-

1. Categories of Services
2. World Trade in Services
3. India's Exports
4. India's Imports
5. India's Balance.

Traditionally, both developed and developing countries specialised in the exchange of goods. In recent decades, the emphasis has changed significantly from the facilitation of commodities trade to the facilitation of services trade. Between 1970 and 1990, the international trade in services expanded by an average of 12% each year, according to the UNDP. It increased by more than thrice between 1990 and 2000, from US$ 800 billion in 1990 to US$ 2500 billion in 2000. The percentage of services in 2013 was 66 percent, or 49.9 trillion dollars, of global GDP at current values of $75.6 trillion dollars.

1.4 Trading

There are two types of trading mainly done by a particular country: **Internal trade** and **International Trade**.

- Internal trade is defined as the buying and selling of goods and services between two parties who are located within a country's political and geographical limits. When it comes to international trade, it can be between two countries..
- International commerce provides foreign reserves for the two trading countries while also allowing for currency exchange. Internal trade does not generate any foreign exchange reserves..

1.4.1 Types of International Trade

- Import Trade: It refers to purchase of goods and services by one country from another country.
- Export Trade: It refers to the sale of goods and services by one country to another country.
- Entrpot Trade: Re-export of products and services is another name for it. It refers to the acquisition of commodities from one country and the subsequent sale of those goods to another country after certain processing.

1.4.2 Agriculture

Agriculture is rapidly being depended upon to provide a wide range of important needs, including providing healthy food for 9 billion people by 2050, boosting earnings, and providing environmental services. Agriculture is confronted by accelerating climate change, increased market risk, tightening resource constraints, a growing need for private sector engagement in delivering agricultural public goods, too-slow progress on raising rural incomes in some regions, and too-sluggish progress on providing environmental services. The World Bank Group has raised its response to more than US$7 billion in new aid per year, combining agriculture, water, forestry, and biodiversity in a cross-sectoral and landscape strategy..

1.4.2.1 Sectors in Agriculture

- Easy Bank Loan,Low Interest rate and low tax
- Crop Production
- Animal Production
- Forestry and Logging
- Fishing, Hunting and Trapping
- Support Activities for Agriculture and Forestry

1.4.3 MSMEs

Since the previous fifty years, the micro, small, and medium enterprise sectors have played an important part in the Indian economy. This sector contributes not only to economic development but also to social development by providing a vast number of job possibilities.
MSMEs (micro, small, and medium enterprises) are the growth engines of the Indian economy, accounting for over 30% of the country's gross domestic product (GDP). They are an important part of the supply chain in terms of exports, accounting for around 40% of total exports. MSMEs also play a significant role in job creation, employing around 110 million people across the country. MSMEs are also connected with the rural economy, as more than half of all MSMEs are located in rural India.

1.4.3.1 New Definition for MSMEs

- Micro enterprises Investment of less than 1 crore and turnover less than 5 crore
- Small enterprises Investment of less than '10 crore and turnover less than '50 crore
- Medium enterprisesInvestment of less than '50 crore and turnover less than '250 crore

1.4.3.2 Old Definition of MSMEs

- Micro enterprises Investment of less than 25 lakhs
- Small enterprises Investment of less than 5 crore
- Medium enterprises Investment of less than 10 crore

1.4.3.3 Benefits of MSMEs

- Easy Bank Loan,Low Interest rate and low tax
- More preference than large enterprises
- National Green Tribunal Ease of taking pollution certification, water pollution certification

1.4.4 Exports,Market size and GDP:

There are roughly 6.3 crore MSMEs in India. From 21.21 lakh (2.1 million) units in 2019, the number of registered MSMEs increased by 18.5 percent Y-o-Y to 25.13 lakh (2.5 million) units in 2020. Through

national and international trade, the Indian MSMEs sector contributes roughly 29% to the GDP.
Micro enterprises will account for 22.06 lakh (2.2 million) units in 2020, up from 18.70 lakh (1.8 million) units in 2019, according to data provided by the MSME Minister in the Rajya Sabha, while small enterprise units increased from 2.41 lakh (0.24 million) units to 2.95 lakh (0.29 million) units. During this time, the number of midsized firms barely climbed from 9,403 to 10,981 units.

Chapter *2*

Literature Survey

For a long time, economists have grappled with predicting issues. It has a long and illustrious history. According to Zarnowitz and Lambros (1987), the association between survey-based dispersion and macroeconomic instability is based on the idea that forecasters have greater impact during times of economic volatility. Fair and Shiller (1990) discovered that the performance of economic projections is influenced by the economy's volatility. According to Nalewaik (2011), GDI is a better indicator of economic health than GDP..

Typically, policymakers make judgments in real time based on imperfect knowledge about current economic situations. Many important statistics are delayed and vulnerable to repeated adjustments. Nowcasting models have become more popular methods for mitigating some of these uncertainties, and forecasters at many central banks and other institutions have employed them extensively. ([6]).

This field has recently had a slew of papers published. Forecasters are aware that recession periods would differ from other eras, according to An, Jalles, and Loungain (2018). Behrens, Pierdzioch, and Risse (2018) demonstrated that the veracity of four German research institutes' long-run inflation forecasts cannot be disputed, and that inflation forecasting is efficient. Berge (2018) discovered that household inflation expectations are related to distinct macroeconomic variables than professional inflation expectations..
The validity of applying machine learning to forecast economic variables is investigated in this study. In G7 countries, a comparison of using the ML model and machine learning to anticipate GDP and consumer prices is made. Regardless of the data size, some data with high oscillations and changes in a short period is insufficient for machine learning. Machine learning looks at the data's pattern, therefore data with a trend is suitable for analysis. For comparison, GDP and consumer prices appear to be appropriate.

Chapter *3*

Proposed Work

3.1 Economic Forecasting and Machine Learning

[5]Economic forecasting is now necessary in almost every industry to assess the potential implications of a financial policy. From business industries to government officials, such policy forecasts are essential. Microeconomics and macroeconomics forecasting use a large quantity of data based on numerous parameters to try to figure out the best relevant variables for a certain sector. Macroeconomic forecasting is concerned with the state of the economy in individual countries and the global economy as a whole. We'll look at one such prediction variable in this chapter: the Gross Domestic Product (GDP) of a few countries.

Machine Learning has risen in popularity over the last decade because to its quick data modelling and superior out-of-the-model predictions. Economists have been experimenting with these models, however there has been some scepticism about the significance of Machine Learning in Macroeconomic Forecasting and its application in the literature.[3].

3.2 Need for Macroeconomic Forecasting

The current economic landscape aids governments in prioritising the work that needs to be done in the short term to recover the economy and plan for economic downturns. Macroeconomic forecasting forecasts the consequences of the recent recession in rich countries (European Union) and developing countries (BRICS), as well as the route forward for developing countries. Predictions are made using production-based metrics such as production, imports, exports, and purchasing power parity (PPP), as well as per capita income.

3.3 Issues faced by Traditional Systems

Based on references from [1] [7] [4] [2] we find that the main backslash faced by traditional systems are as follows:

- Non-Linearity Fitting fundamental parameters to higher dimensional data sets may not fit well, resulting in larger predicting error. In this case, we can utilise the kernel method, which involves using distinct functions of parameters to reduce predicting error.
- Dimensionality Reduction (Question the relevance of this part.) When working with high-dimensional data sets, it's difficult to eliminate unimportant parameters, and most traditional models don't provide a well-established solution.
- Loss or Error Function The OLS Error isn't appropriate for all models, and selecting the right error for the right model appears to be a challenge when working with forecasting models.
- Prediction Relevance with increased amount of data The accuracy and relevance of predictors using existing AIC,BIC do not produce good enough accurate models utilising the abundant sources of data as the age of data expands.

3.4 Traditional Systems for Forecasting

Co-linearity, dimensionality, predictor relevance, and non-linearity are all challenges that techniques based on OLS errors struggle to overcome. As a result, even the most advanced forecasting models frequently produce huge forecast mistakes, particularly when the variable to be anticipated is volatile, as it is in many emerging market and developing nations. Because they emphasise out-of-sample (rather than in-sample) performance and better handle nonlinear interactions among a large number of predictors, ML models can outperform traditional forecasting methods.

3.4.1 Auto Regressive Moving Average Models ARIMA

We strive to improve our models by incorporating some features from Auto Regressive Time Series Classification Models with Moving Average Models in Auto Regressive Integrated Moving Average Models.
Some of the Equations to describe the model are : -
$Y_t = \alpha_t * (1 - B)^i + \epsilon_t + \beta_t * e_t ....$
Where α_t is the input at time t,
and ϵ_t is the noise,
B*$\alpha_t = \alpha_{t-1}$ is what defines the function B it is the relation between input at time t and input at time t-1,
The value of i in the equation determines the Order of the Time Series Model,
β_t is the moving average constant along with the moving average function,

Pros:- Are easy to learn and implement when normalized data using stationary time-series is used.

Cons:- Non-Linearity and hence cannot be fitted accurately in most models.

3.4.2 Dynamic Factor Models

An important motivation for considering DFMs is that, if one knew the factors are Gaussian, then one can make efficient forecasts for an individual variable using the population regression of that variable on the lagged factors of that variable.

$x_{kt} = \lambda_i(L) * f(t) + e_{it}$

$y_{t+h} = \lambda_Y(L) * f(t) + e_{yt}$

$y_{t+h} = \beta(L) * f(t) + \gamma(L) * Y_t + \epsilon_{t+h}$

Legends : x_{kt} λ is the lag-order polynomial function with input L . e_{it} is the noise/disturbance term.

- Pros:

  Gaussaian models or similar models work well

  Takes care of latent variables

- Cons(Shortcomings):

  Non-Linearity Using something similar to the kernel trick common with machine learning are difficult to implement.

  Predictor Relevance In high dimensionality problems the irrelevant predictors are not easy to detect and can cause the model to allow for higher noise in prediction.

3.5 Data Rich Vs Data Poor Conditions

Prediction variables are assumed to be Identically Idependent Distributed (IID). When the number of predictors exceeds the number of observations in a dataset, it is referred to be ”Data Poor.” If the number of predictors is smaller than the number of observations in a dataset, it is referred to be ”Data Rich.” In our comparison of Data Rich (Developed and some Developing Countries) and Data Poor (Developing and Underdeveloped Countries), the various sorts of methods that can be used to deal with both types of datasets will be explored in this part and the following section. [8]

Chapter *4*

Simulation and Results

4.1 Libraries Used in Codes

```
import pandas as pd
import numpy as np
import matplotlib
import seaborn  as sns
import statistics
import scipy
from scipy import stats
import matplotlib.pyplot as plt
```

```
#Testing and Training a Model for Predicting India's GDP using Linear Regression
from sklearn import preprocessing, svm
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import seaborn as sns
from matplotlib import pyplot as plt
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, mean_squared_log_error
```

Figure 4.1 Libraries

4.2 Dataset

```
gdp = pd.read_csv("gdp_csv.csv")
```

```
data = pd.read_csv('input.csv',decimal=',')
```

Figure 4.2 Data-set

4.3 Code 1

4.3.1 Table Description

```
print(gdp.shape)
```

```
(11507, 4)
```

```
gdp.head()
```

	Country Name	Country Code	Year	Value
0	Arab World	ARB	1968	2.576068e+10
1	Arab World	ARB	1969	2.843420e+10
2	Arab World	ARB	1970	3.138550e+10
3	Arab World	ARB	1971	3.642691e+10
4	Arab World	ARB	1972	4.331606e+10

```
gdp.tail()
```

	Country Name	Country Code	Year	Value
11502	Zimbabwe	ZWE	2012	1.424249e+10
11503	Zimbabwe	ZWE	2013	1.545177e+10
11504	Zimbabwe	ZWE	2014	1.589105e+10
11505	Zimbabwe	ZWE	2015	1.630467e+10
11506	Zimbabwe	ZWE	2016	1.661996e+10

```
gdp.describe()
```

	Year	Value
count	11507.000000	1.150700e+04
mean	1991.265230	1.005972e+12
std	15.886648	4.533056e+12
min	1960.000000	8.824448e+06
25%	1978.000000	2.056874e+09
50%	1993.000000	1.436880e+10
75%	2005.000000	1.796394e+11
max	2016.000000	7.904923e+13

Figure 4.3 In this first output shows total number of rows and columns and rest two denotes the top 5 and bottom 5 entries of the data sets.

4.3.2 country Name

```
gdp["Country Name"].unique()
```

```
array(['Arab World', 'Caribbean small states',
       'Central Europe and the Baltics', 'Early-demographic dividend',
       'East Asia & Pacific',
       'East Asia & Pacific (excluding high income)',
       'East Asia & Pacific (IDA & IBRD countries)', 'Euro area',
       'Europe & Central Asia',
       'Europe & Central Asia (excluding high income)',
       'Europe & Central Asia (IDA & IBRD countries)', 'European Union',
       'Fragile and conflict affected situations',
       'Heavily indebted poor countries (HIPC)', 'High income',
       'IBRD only', 'IDA & IBRD total', 'IDA blend', 'IDA only',
       'IDA total', 'Late-demographic dividend',
       'Latin America & Caribbean',
       'Latin America & Caribbean (excluding high income)',
       'Latin America & the Caribbean (IDA & IBRD countries)',
       'Least developed countries: UN classification',
       'Low & middle income', 'Low income', 'Lower middle income',
       'Middle East & North Africa',
       'Middle East & North Africa (excluding high income)',
       'Middle East & North Africa (IDA & IBRD countries)',
       'Middle income', 'North America', 'OECD members',
       'Other small states', 'Pacific island small states',
       'Post-demographic dividend', 'Pre-demographic dividend',
       'Small states', 'South Asia', 'South Asia (IDA & IBRD)',
       'Sub-Saharan Africa', 'Sub-Saharan Africa (excluding high income)',
       'Sub-Saharan Africa (IDA & IBRD countries)', 'Upper middle income',
       'World', 'Afghanistan', 'Albania', 'Algeria', 'American Samoa',
       'Andorra', 'Angola', 'Antigua and Barbuda', 'Argentina', 'Armenia',
       'Aruba', 'Australia', 'Austria', 'Azerbaijan', 'Bahamas, The',
       'Bahrain', 'Bangladesh', 'Barbados', 'Belarus', 'Belgium',
       'Belize', 'Benin', 'Bermuda', 'Bhutan', 'Bolivia',
       'Bosnia and Herzegovina', 'Botswana', 'Brazil',
       'Brunei Darussalam', 'Bulgaria', 'Burkina Faso', 'Burundi',
       'Cabo Verde', 'Cambodia', 'Cameroon', 'Canada', 'Cayman Islands',
       'Central African Republic', 'Chad', 'Channel Islands', 'Chile',
       'China', 'Colombia', 'Comoros', 'Congo, Dem. Rep.', 'Congo, Rep.',
       'Costa Rica', "Cote d'Ivoire", 'Croatia', 'Cuba', 'Cyprus',
       'Czech Republic', 'Denmark', 'Djibouti', 'Dominica',
       'Dominican Republic', 'Ecuador', 'Egypt, Arab Rep.', 'El Salvador',
       'Equatorial Guinea', 'Eritrea', 'Estonia', 'Ethiopia',
       'Faroe Islands', 'Fiji', 'Finland', 'France', 'French Polynesia',
       'Gabon', 'Gambia, The', 'Georgia', 'Germany', 'Ghana', 'Greece',
       'Greenland', 'Grenada', 'Guam', 'Guatemala', 'Guinea',
```

Figure 4.4 Names of the country in data sets and also different region like Arabic region etc.

4.3.3 Description for India and USA

```
#for India
ind = gdp[gdp['Country Name']=='India']
print(ind)
#Shape
print(gdp.shape)
```

```
     Country Name Country Code  Year         Value
6074        India          IND  1960  3.653593e+10
6075        India          IND  1961  3.870910e+10
6076        India          IND  1962  4.159907e+10
6077        India          IND  1963  4.777600e+10
6078        India          IND  1964  5.572687e+10
6079        India          IND  1965  5.876042e+10
6080        India          IND  1966  4.525364e+10
6081        India          IND  1967  4.946617e+10
6082        India          IND  1968  5.237732e+10
6083        India          IND  1969  5.766833e+10
6084        India          IND  1970  6.158980e+10
6085        India          IND  1971  6.645256e+10
6086        India          IND  1972  7.050991e+10
6087        India          IND  1973  8.437454e+10
6088        India          IND  1974  9.819828e+10
6089        India          IND  1975  9.715922e+10
6090        India          IND  1976  1.013470e+11
6091        India          IND  1977  1.198667e+11
6092        India          IND  1978  1.354688e+11
6093        India          IND  1979  1.509508e+11
6094        India          IND  1980  1.838399e+11
6095        India          IND  1981  1.909095e+11
6096        India          IND  1982  1.980377e+11
6097        India          IND  1983  2.153508e+11
6098        India          IND  1984  2.093282e+11
6099        India          IND  1985  2.294103e+11
6100        India          IND  1986  2.456647e+11
6101        India          IND  1987  2.753114e+11
6102        India          IND  1988  2.926327e+11
6103        India          IND  1989  2.920933e+11
6104        India          IND  1990  3.166973e+11
6105        India          IND  1991  2.665023e+11
6106        India          IND  1992  2.843639e+11
6107        India          IND  1993  2.755704e+11
```

```
ind.plot(x='Year', y='Value', kind='line')
plt.show()
```

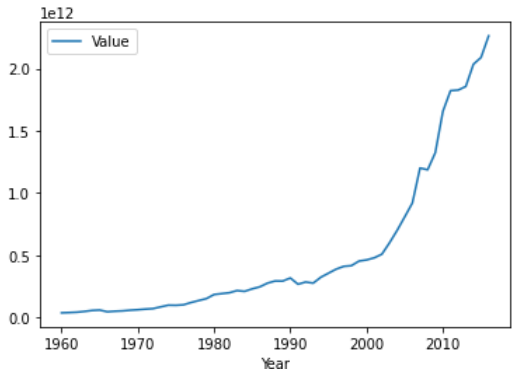


Figure 4.5 GDP values of India since 1968 to 2016

```
#For USA
usa = gdp[gdp['Country Name']=='United States']
print(usa)
```

```
       Country Name Country Code  Year         Value
11029  United States          USA  1960  5.433000e+11
11030  United States          USA  1961  5.633000e+11
11031  United States          USA  1962  6.051000e+11
11032  United States          USA  1963  6.386000e+11
11033  United States          USA  1964  6.858000e+11
11034  United States          USA  1965  7.437000e+11
11035  United States          USA  1966  8.150000e+11
11036  United States          USA  1967  8.617000e+11
11037  United States          USA  1968  9.425000e+11
11038  United States          USA  1969  1.019900e+12
11039  United States          USA  1970  1.075884e+12
11040  United States          USA  1971  1.167770e+12
11041  United States          USA  1972  1.282449e+12
11042  United States          USA  1973  1.428549e+12
11043  United States          USA  1974  1.548825e+12
11044  United States          USA  1975  1.688923e+12
11045  United States          USA  1976  1.877587e+12
11046  United States          USA  1977  2.085951e+12
11047  United States          USA  1978  2.356571e+12
11048  United States          USA  1979  2.632143e+12
11049  United States          USA  1980  2.862505e+12
11050  United States          USA  1981  3.210956e+12
11051  United States          USA  1982  3.344991e+12
11052  United States          USA  1983  3.638137e+12
11053  United States          USA  1984  4.040693e+12
11054  United States          USA  1985  4.346734e+12
11055  United States          USA  1986  4.590155e+12
11056  United States          USA  1987  4.870217e+12
11057  United States          USA  1988  5.252629e+12
11058  United States          USA  1989  5.657693e+12
11059  United States          USA  1990  5.979589e+12
11060  United States          USA  1991  6.174043e+12
11061  United States          USA  1992  6.539299e+12
11062  United States          USA  1993  6.878718e+12
11063  United States          USA  1994  7.308755e+12
11064  United States          USA  1995  7.664060e+12
```

```
usa.plot(x='Year', y='Value', kind='line')
plt.show()
```

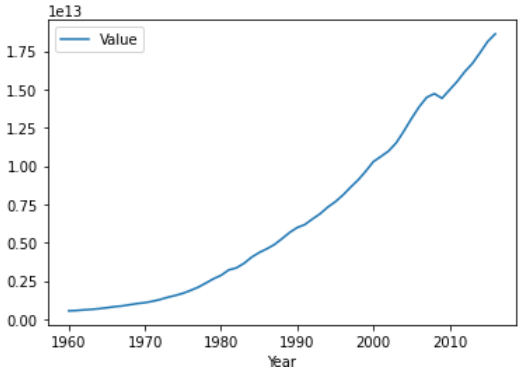


Figure 4.6 GDP values of USA since 1968 to 2016

4.3.4 Splitting into Training and Test set

```
# x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.10,random_state =
x_train = x[:int(x.shape[0]*0.8)]
x_test = x[int(x.shape[0]*0.8):]
y_train = y[:int(y.shape[0]*0.8)]
y_test = y[int(y.shape[0]*0.8):]
```

Figure 4.7 Splitting the Data into training set and test set in the ratio of 80:20

4.3.5 Classification

Linear Regression

```
regr = LinearRegression()

regr.fit(x_train, y_train)
print('coefficient of determination:',regr.score(x_test, y_test))
```

```
coefficient of determination: -4.676407010020001
```

Figure 4.8 Calling the Linear Regression Method

```
# Data scatter of predicted values
regr_y_test_pred = regr.predict(x_test)
print('predicted response:', regr_y_test_pred, sep='\n')
plt.scatter(x_test, y_test, color ='b')
plt.plot(x_test, regr_y_test_pred, color ='k')
plt.show()
```

```
predicted response:
[[5.04329521e+11]
 [5.16557608e+11]
 [5.28785695e+11]
 [5.41013781e+11]
 [5.53241868e+11]
 [5.65469954e+11]
 [5.77698041e+11]
 [5.89926128e+11]
 [6.02154214e+11]
 [6.14382301e+11]
 [6.26610387e+11]
 [6.38838474e+11]]
```

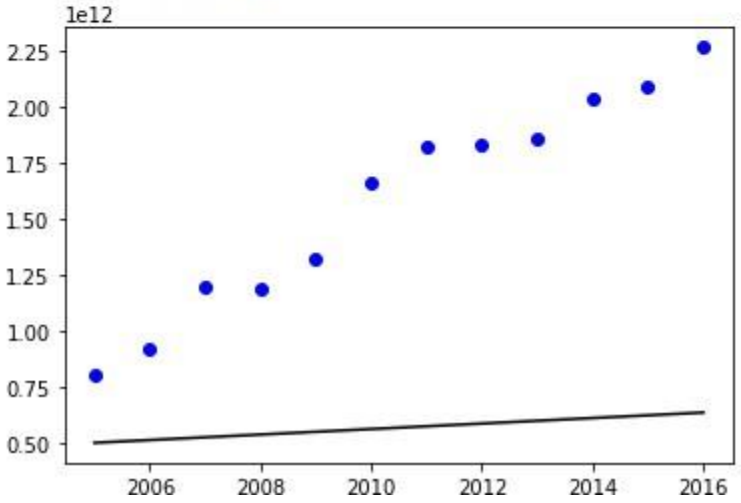


Figure 4.9 Calling the Linear Regression Method

```
regr_y_pred = regr.predict([[2017]])
regr_y_pred
```

```
array([[6.51066561e+11]])
```

Figure 4.10 Predicting the value from Linear Regression Method

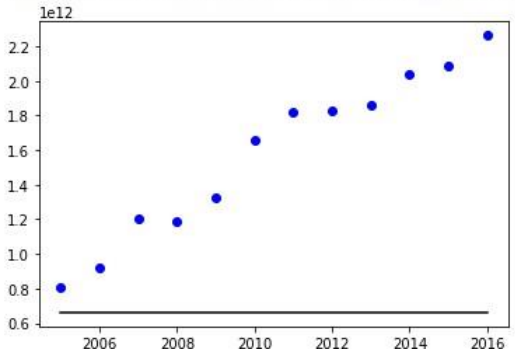
Random Forest

```
#Fitting Decision Tree classifier to the training set
from sklearn.ensemble import RandomForestRegressor
rfr = RandomForestRegressor()
rfr.fit(x_train, y_train)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:4: DataConversionWarning: A column-vector y was passed when a 1d array
  after removing the cwd from sys.path.
RandomForestRegressor()
```

```
rfr_y_test_pred = rfr.predict(x_test)
print('predicted response:', rfr_y_test_pred, sep='\n')
plt.scatter(x_test, y_test, color ='b')
plt.plot(x_test, rfr_y_test_pred, color ='k')
plt.show()
```

```
predicted response:
[6.61502176e+11 6.61502176e+11 6.61502176e+11 6.61502176e+11
 6.61502176e+11 6.61502176e+11 6.61502176e+11 6.61502176e+11
 6.61502176e+11 6.61502176e+11 6.61502176e+11 6.61502176e+11]
```




```
rfr_y_pred = rfr.predict([[2017]])
rfr_y_pred
```

```
array([6.61502176e+11])
```

Figure 4.11 Predicting the value from Random forest Method

Custom made Method in which Mean of values of linear regression and random forest is taken

```
custom_model_y_pred = np.zeros((regr_y_test_pred.shape[0],1))
for i in range(regr_y_test_pred.shape[0]):
  custom_model_y_pred[i] = (rfr_y_test_pred[i]+regr_y_test_pred[i])/2
print('predicted response:', custom_model_y_pred, sep='\n')
plt.scatter(x_test, y_test, color ='b')
plt.plot(x_test, custom_model_y_pred, color ='k')
plt.show()
```

```
predicted response:
[[5.82915848e+11]
 [5.89029892e+11]
 [5.95143935e+11]
 [6.01257978e+11]
 [6.07372022e+11]
 [6.13486065e+11]
 [6.19600108e+11]
 [6.25714152e+11]
 [6.31828195e+11]
 [6.37942238e+11]
 [6.44056282e+11]
 [6.50170325e+11]]
```

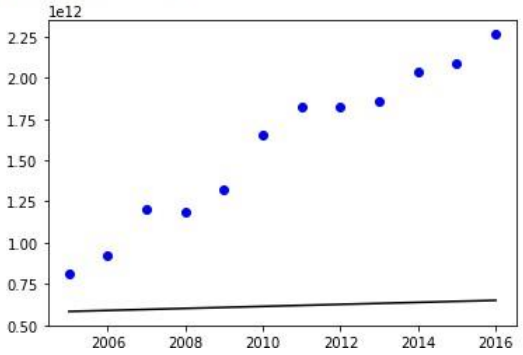


Figure 4.12 Predicting the value from Custom Made Method

Similarly, We can found the GDP for all the Countries as given in the Data-sets from Code 1.

4.4 Code 2

4.4.1 Table Description

```
print('number of missing data:')
print(data.isnull().sum())
data.describe(include='all')
```

```
number of missing data:
Country                0
Region                 0
Population             0
GDP ($ per capita)     1
Literacy (%)          18
Agriculture           15
Industry              16
Service               15
dtype: int64
```

	Country	Region	Population	GDP ($ per capita)	Literacy (%)	Agriculture	Industry	Service
count	227	227	2.270000e+02	226.000000	209.000000	212.000000	211.000000	212.000000
unique	227	11	NaN	NaN	NaN	NaN	NaN	NaN
top	South Africa	SUB-SAHARAN AFRICA	NaN	NaN	NaN	NaN	NaN	NaN
freq	1	51	NaN	NaN	NaN	NaN	NaN	NaN
mean	NaN	NaN	2.874028e+07	9689.823009	82.838278	0.150844	0.282711	0.565283
std	NaN	NaN	1.178913e+08	10049.138513	19.722173	0.146798	0.138272	0.165841
min	NaN	NaN	7.026000e+03	500.000000	17.600000	0.000000	0.020000	0.062000
25%	NaN	NaN	4.376240e+05	1900.000000	70.600000	0.037750	0.193000	0.429250
50%	NaN	NaN	4.786994e+06	5550.000000	92.500000	0.099000	0.272000	0.571000
75%	NaN	NaN	1.749777e+07	15700.000000	98.000000	0.221000	0.341000	0.678500
max	NaN	NaN	1.313974e+09	55100.000000	100.000000	0.769000	0.906000	0.954000

```
data.groupby('Region')[['GDP ($ per capita)','Literacy (%)','Agriculture']].median()
```

Region	GDP ($ per capita)	Literacy (%)	Agriculture
ASIA (EX. NEAR EAST)	3450.0	90.60	0.1610
BALTICS	11400.0	99.80	0.0400
C.W. OF IND. STATES	3450.0	99.05	0.1980
EASTERN EUROPE	9100.0	98.60	0.0815
LATIN AMER. & CARIB	6300.0	94.05	0.0700
NEAR EAST	9250.0	83.00	0.0350
NORTHERN AFRICA	6000.0	70.00	0.1320
NORTHERN AMERICA	29800.0	97.50	0.0100
OCEANIA	5000.0	95.00	0.1505
SUB-SAHARAN AFRICA	1300.0	62.95	0.2760
WESTERN EUROPE	27200.0	99.00	0.0220

Figure 4.13 Table description

4.4.2 Region with Codes

```
LE = LabelEncoder()
data['Region_label'] = LE.fit_transform(data['Region'])
data.groupby(Region')
```

	Country	Region	Population	GDP ($ per capita)	Literacy (%)	Agriculture	Industry	Service	Region_label
0	Afghanistan	ASIA (EX. NEAR EAST)	31056997	700.0	36.0	0.380	0.240	0.380	0
1	Albania	EASTERN EUROPE	3581655	4500.0	86.5	0.232	0.188	0.579	3
2	Algeria	NORTHERN AFRICA	32930091	6000.0	70.0	0.101	0.600	0.298	6
3	American Samoa	OCEANIA	57794	8000.0	97.0	NaN	NaN	NaN	8
4	Andorra	WESTERN EUROPE	71201	19000.0	100.0	NaN	NaN	NaN	10

Figure 4.14 Table description

4.4.3 GDP Plot

```
fig, ax = plt.subplots(figsize=(16,6))
#ax = fig.add_subplot(111)
top_gdp_countries = data.sort_values('GDP ($ per capita)',ascending=False).head(20)
mean = pd.DataFrame({'Country':['World mean'], 'GDP ($ per capita)':[data['GDP ($ per capita)'].mean()]})
gdps = pd.concat([top_gdp_countries[['Country','GDP ($ per capita)']],mean],ignore_index=True)

sns.barplot(x='Country',y='GDP ($ per capita)',data=gdps, palette='Set3')
ax.set_xlabel(ax.get_xlabel(),labelpad=15)
ax.set_ylabel(ax.get_ylabel(),labelpad=30)
ax.xaxis.label.set_fontsize(16)
ax.yaxis.label.set_fontsize(16)
plt.xticks(rotation=90)
plt.show()
```

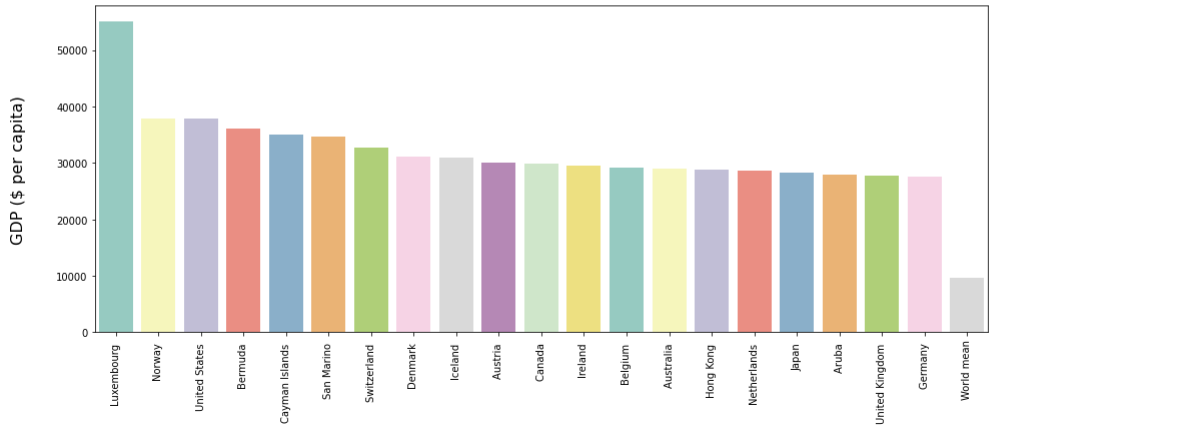


Figure 4.15 GDP plot

4.4.4 Correlation

```
plt.figure(figsize=(16,12))
sns.heatmap(data=data.iloc[:,2:].corr(),annot=True,fmt='.2f',cmap='coolwarm')
plt.show()
```

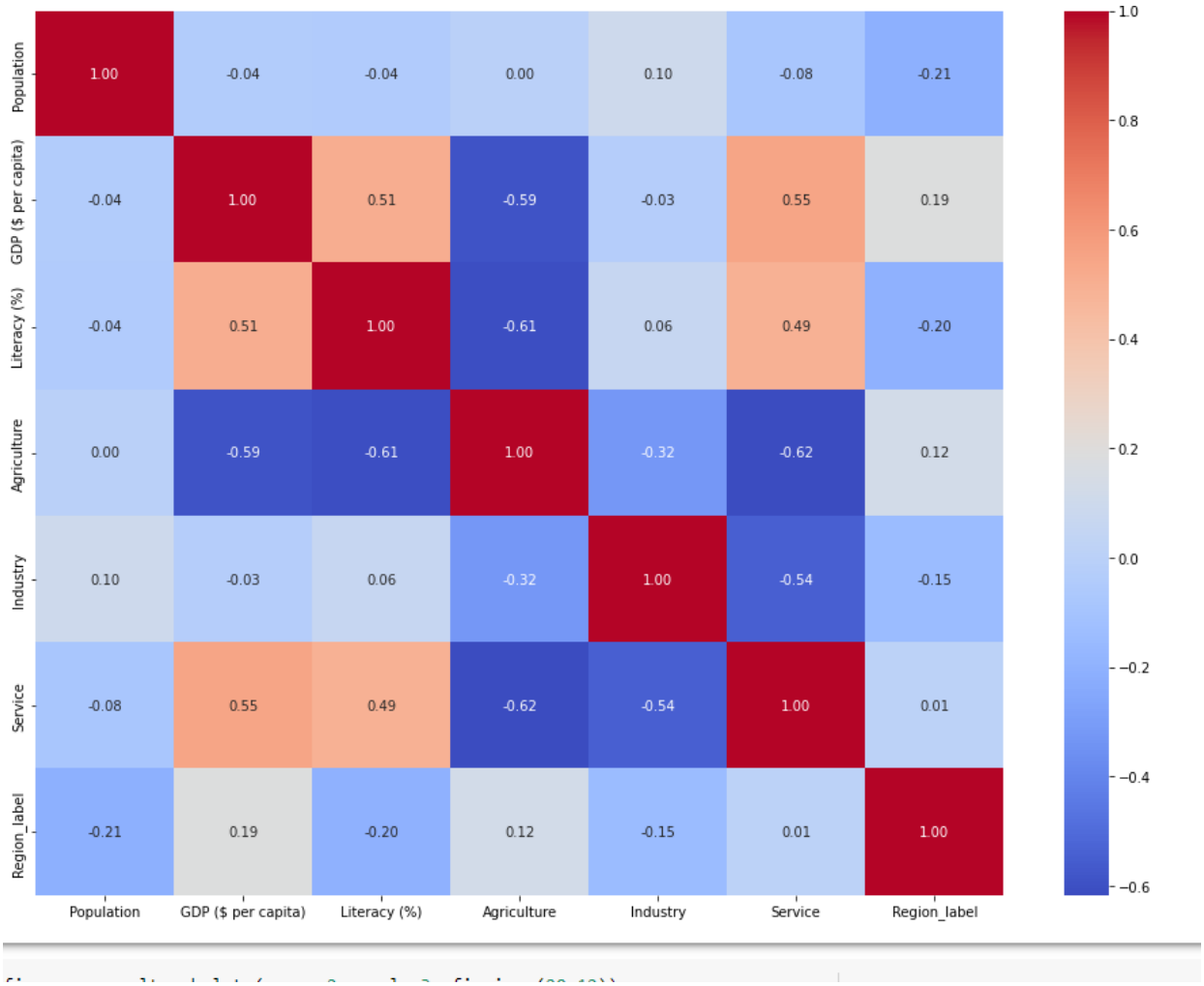


Figure 4.16 heat map:Correlation between Variables

```
fig, axes = plt.subplots(nrows=2, ncols=3, figsize=(20,12))
plt.subplots_adjust(hspace=0.4)

corr_to_gdp = pd.Series()
for col in data.columns.values[2:]:
    if ((col!='GDP ($ per capita)')&(col!='Climate')):
        corr_to_gdp[col] = data['GDP ($ per capita)'].corr(data[col])
abs_corr_to_gdp = corr_to_gdp.abs().sort_values(ascending=False)
corr_to_gdp = corr_to_gdp.loc[abs_corr_to_gdp.index]

for i in range(2):
    for j in range(3):
        sns.regplot(x=corr_to_gdp.index.values[i*3+j], y='GDP ($ per capita)', data=data, ax=axes[i,j], fit_reg=False, marker='.')
        title = 'correlation='+str(corr_to_gdp[i*3+j])
        axes[i,j].set_title(title)
axes[1,2].set_xlim(0,102)
plt.show()
```

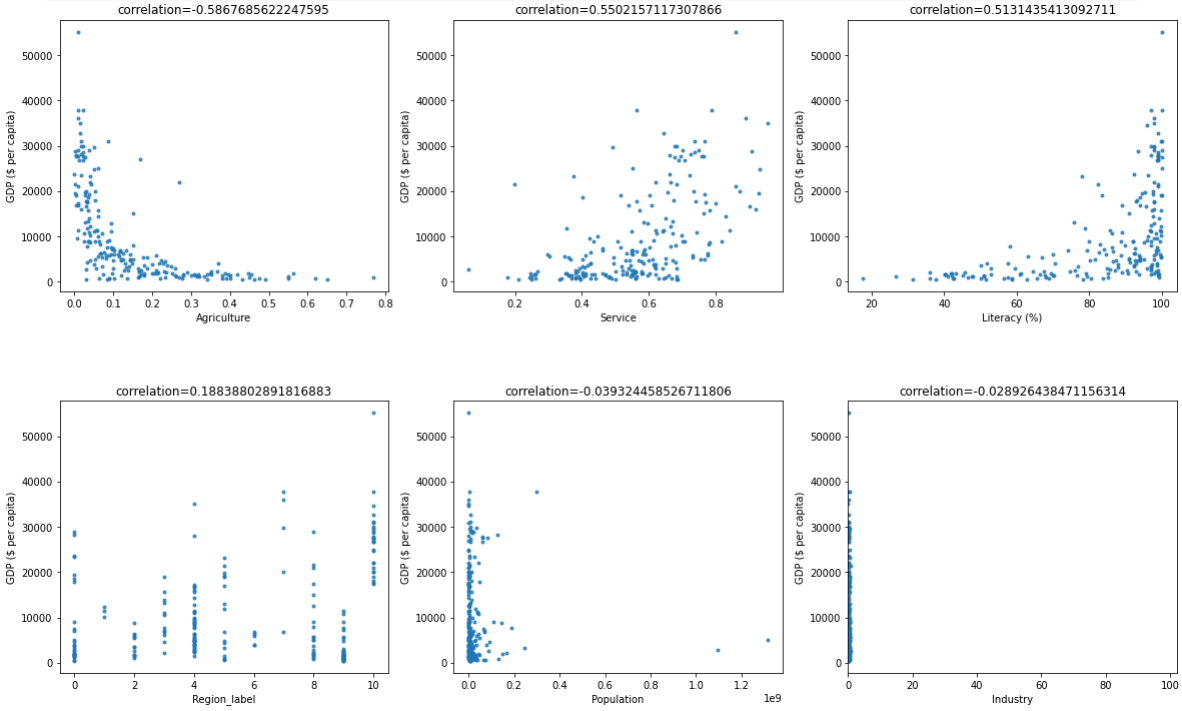


Figure 4.17 Correlation between Variables

4.4.5 Splitting into Training and Test set

```
training_features = ['Region','Population',
                     'GDP ($ per capita)','Literacy (%)',
                     'Agriculture','Industry',
                     'Service']
target = 'GDP ($ per capita)'
X = data[training_features]
Y = data[target]

X = X.apply(pd.to_numeric, errors='coerce')
Y = Y.apply(pd.to_numeric, errors='coerce')
X.fillna(0, inplace=True)
Y.fillna(0, inplace=True)
np.nan_to_num(X)
np.nan_to_num(Y)

train_X, test_X,train_Y,test_Y = train_test_split(X,Y, test_size=0.3, shuffle=True)
train_X.head()
```

	Region	Population	GDP ($ per capita)	Literacy (%)	Agriculture	Industry	Service
130	0.0	60422	1600.0	93.7	0.317	0.149	0.534
206	0.0	70413958	6700.0	86.5	0.117	0.298	0.585
5	0.0	12127071	1900.0	42.0	0.096	0.658	0.246
100	0.0	6352117	19800.0	95.4	0.026	0.317	0.657
94	0.0	1095351995	2900.0	59.5	0.186	0.276	0.538

Figure 4.18 Splitting data-set into training set and test set in ratio of 80:20

4.4.6 Classification

Linear Regression

```
model = LinearRegression()
model.fit(train_X, train_Y)
train_pred_Y = model.predict(train_X)
test_pred_Y = model.predict(test_X)
train_pred_Y = pd.Series(train_pred_Y.clip(0, train_pred_Y.max()), index=train_Y.index)
test_pred_Y = pd.Series(test_pred_Y.clip(0, test_pred_Y.max()), index=test_Y.index)

rmse_train = np.sqrt(mean_squared_error(train_pred_Y, train_Y))
msle_train = mean_squared_log_error(train_pred_Y, train_Y)
rmse_test = np.sqrt(mean_squared_error(test_pred_Y, test_Y))
msle_test = mean_squared_log_error(test_pred_Y, test_Y)

print('rmse_train:',rmse_train,'msle_train:',msle_train)
print('rmse_test:',rmse_test,'msle_test:',msle_test)
```

```
rmse_train: 3.2798065949303284e-12 msle_train: 2.61622224263171e-30
rmse_test: 3.986587745264883e-12 msle_test: 2.4580448496017035e-30
```

Figure 4.19 Output denotes the root mean square error for predicted values and respective log values

Random Forest

```
model = RandomForestRegressor(n_estimators = 50,
                              max_depth = 6,
                              min_weight_fraction_leaf = 0.05,
                              max_features = 0.8,
                              random_state = 42)
model.fit(train_X, train_Y)
train_pred_Y = model.predict(train_X)
test_pred_Y = model.predict(test_X)
train_pred_Y = pd.Series(train_pred_Y.clip(0, train_pred_Y.max()), index=train_Y.index)
test_pred_Y = pd.Series(test_pred_Y.clip(0, test_pred_Y.max()), index=test_Y.index)

rmse_train = np.sqrt(mean_squared_error(train_pred_Y, train_Y))
msle_train = mean_squared_log_error(train_pred_Y, train_Y)
rmse_test = np.sqrt(mean_squared_error(test_pred_Y, test_Y))
msle_test = mean_squared_log_error(test_pred_Y, test_Y)

print('rmse_train:',rmse_train,'msle_train:',msle_train)
print('rmse_test:',rmse_test,'msle_test:',msle_test)
```

```
rmse_train: 1912.840765656693 msle_train: 0.4076492150272322
rmse_test: 1217.2312300955523 msle_test: 0.02375745907403671
```

Figure 4.20 Output denotes the root mean square error for predicted values and respective log values

```
plt.figure(figsize=(18,12))

train_test_Y = train_Y.append(test_Y)
train_test_pred_Y = train_pred_Y.append(test_pred_Y)

data_shuffled = data.loc[train_test_Y.index]
label = data_shuffled['Country']

colors = {'ASIA (EX. NEAR EAST)         ':'red',
          'EASTERN EUROPE                     ':'orange',
          'NORTHERN AFRICA                    ':'gold',
          'OCEANIA                            ':'green',
          'WESTERN EUROPE                     ':'blue',
          'SUB-SAHARAN AFRICA                 ':'purple',
          'LATIN AMER. & CARIB    ':'olive',
          'C.W. OF IND. STATES ':'cyan',
          'NEAR EAST                          ':'hotpink',
          'NORTHERN AMERICA                   ':'lightseagreen',
          'BALTICS                            ':'rosybrown'}

for region, color in colors.items():
    X = train_test_Y.loc[data_shuffled['Region']==region]
    Y = train_test_pred_Y.loc[data_shuffled['Region']==region]
    ax = sns.regplot(x=X, y=Y, marker='.', fit_reg=False, color=color, scatter_kws={'s':200, 'linewidths':0}, label=region)
plt.legend(loc=4,prop={'size': 12})

ax.set_xlabel('GDP ($ per capita) ground truth',labelpad=40)
ax.set_ylabel('GDP ($ per capita) predicted',labelpad=40)
ax.xaxis.label.set_fontsize(24)
ax.yaxis.label.set_fontsize(24)
ax.tick_params(labelsize=12)

x = np.linspace(-1000,50000,100) # 100 linearly spaced numbers
y = x
plt.plot(x,y,c='gray')

plt.xlim(-1000,60000)
plt.ylim(-1000,40000)
for i in range(0,train_test_Y.shape[0]):
    if((data_shuffled['Population'].iloc[i]>1e8) |
       (data_shuffled['GDP ($ per capita)'].iloc[i]>10000)):
        plt.text(train_test_Y.iloc[i]+200, train_test_pred_Y.iloc[i]-200, label.iloc[i], size='small')
```

Figure 4.21

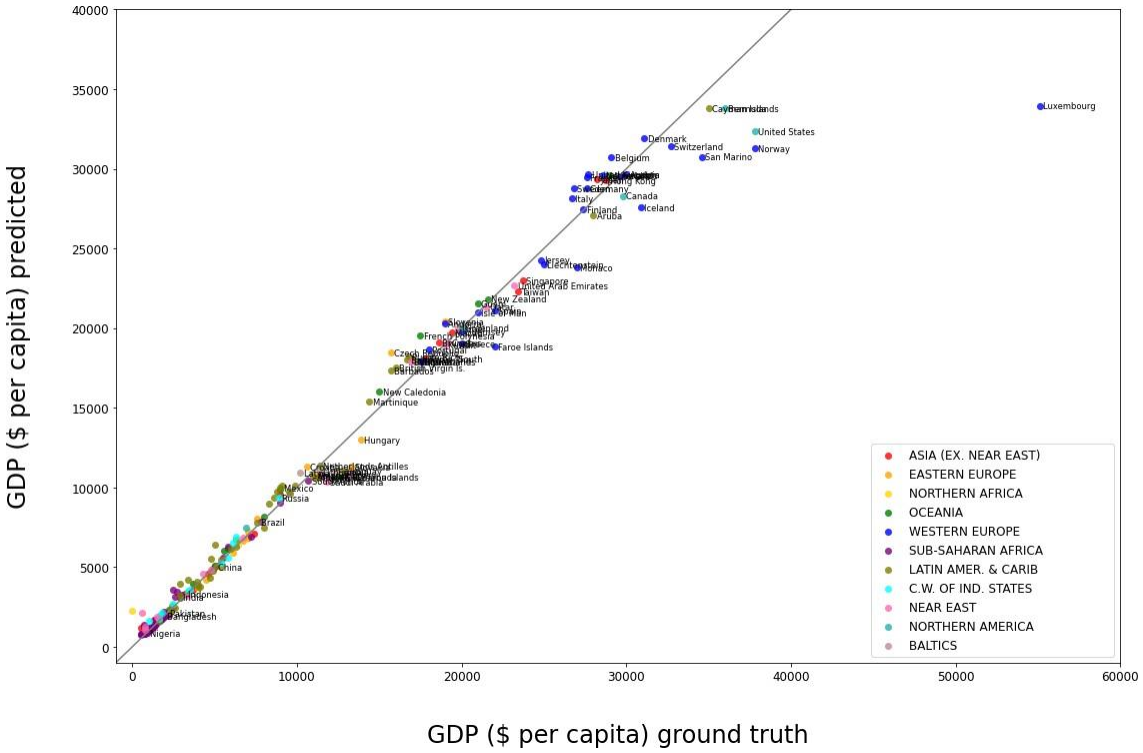


Figure 4.22 Output denotes the predicted values

Chapter *5*

Conclusions and Future Work

In this project we uses machine learning models to predict the GDP ($) for India and United States of America . We found out that random forest suits better than decision tree because accuracy is more in case of random forest

5.1 Scope of further work

The following project was for predicting GDP but we can used for other variables such as inflation. Also we use these models to predict stock price predcition ,post omricon prediction .

Bibliography

[1] Philippe Goulet Coulombe et al. “How is machine learning useful for macroeconomic forecasting?” In: *arXiv preprint arXiv:2008.12477* (2020).

[2] Pradyot Ranjan Jena et al. “Impact of COVID-19 on GDP of major economies: Application of the artificial neural network forecaster”. In: *Economic Analysis and Policy* 69 (2021), pp. 324–339. ISSN: 0313-5926. DOI: https://doi.org/10.1016/j.eap.2020.12.013. URL: https://www.sciencedirect.com/science/article/pii/S0313592620304604.

[3] Sendhil Mullainathan and Jann Spiess. “Machine Learning: An Applied Econometric Approach”. In: *Journal of Economic Perspectives* 31.2 (2017), pp. 87–106. DOI: 10.1257/jep.31.2.87. URL: https://www.aeaweb.org/articles?id=10.1257/jep.31.2.87.

[4] Sendhil Mullainathan and Jann Spiess. “Machine Learning: An Applied Econometric Approach”. In: *Journal of Economic Perspectives* 31.2 (2017), pp. 87–106. DOI: 10.1257/jep.31.2.87. URL: https://www.aeaweb.org/articles?id=10.1257/jep.31.2.87.

[5] Eric Qian. “Nowcasting Indian GDP with Google year=2021 ,Search data”. In: (2021). DOI: https://doi.org/10.17615/wq3c- kc90. URL: https://www.aeaweb.org/articles?id=10.1257/jep.31.2.87.

[6] Adam Richardson, Thomas van Florenstein Mulder, and Tuğrul Vehbi. “Nowcasting GDP using machine-learning algorithms: A real-time assessment”. In: *International Journal of Forecasting* 37.2 (2021), pp. 941–948. ISSN: 0169-2070. DOI: https : / / doi . org / 10 . 1016 / j . ijforecast.2020.10.005. URL: https://www.sciencedirect.com/science/article/pii/S016920702030159X.

[7] Mark W Watson. “Vector autoregressions and cointegration”. In: *Handbook of econometrics* 4 (1994), pp. 2843–2915.

[8] Jaehyun Yoon. “Forecasting of Real GDP Growth Using Machine Learning Models: Gradient Boosting and Random Forest Approach”. In: *Computational Economics* (2021). DOI: https://doi.org/10.1007/s10614-020-10054-w.