INDUSTRIAL & SYSTEMS ENGINEERING
TEXAS A&M UNIVERSITY

Gross Domestic Product of India (1960-2017)

ISEN 613

ENGINEERING DATA ANALYSIS PROJECT REPORT

Submitted by:	UIN:	Contribution:
ARCH DESAI	627006997	100%
ISHITA BHARGAVA	827004766	100%
KRISIL PATEL	527003189	100%
SMRITI MEHTA	527005062	100%
VATSAL THAKKAR	128003914	100%

ACKNOWLEDGEMENT

We as a team would like to firstly thank Prof. Na Zou, for assigning us such a challenging and holistic project. It is through this project we were able to apply our theoretical concepts and knowledge learned in the course and see its wide scale application in real-world problem solving.

Secondly, we would also like to thank Mr. Ming Li, teaching assistant and grader for our subject, for his constant guidance throughout the project.

Thirdly, we would like to thank Texas A&M University for providing us with the resources and support, without which we would not be able to complete this project.

Through this medium we would also like to acknowledge the fact that this complied project report and the work we are submitting is our original work. We hold the ideals of our school Texas A&M in high regards and we abide by the code of honor.

Table of Contents

List of Figures

List of Tables

1. EXECUTIVE SUMMARY

The Gross Domestic Product (GDP) of an economy is a measure of total production. More precisely, it is the monetary value of all goods and services produced within a country or region in a specific time period. Economic prosperity is measured as via growth domestic product (GDP) per capita, the value of all goods and services produced by a country in one year divided by the country's population. Economic growth is the measure of the change of GDP from one year to the next [1].

India being the fastest growing trillion-dollar economy in the world and the sixth largest with a nominal GDP of $2.61 trillion. The country ranks third when GDP is compared in terms of purchasing power parity at $9.45 trillion. India's growth rate is expected to rise from 6.7% in 2017 to 7.3% in 2018 and 7.5% in 2019, as drags from the currency exchange initiative and the introduction of the goods and services tax fade according to IMF [2].

There are however several indicators that count towards the GDP, and as India changes its economic status from a developing to a developed country, how these factors over the time have influenced GDP over the time is worth for an analysis.

India's post-independence journey began as an agrarian nation, however, over the years manufacturing and services sector have emerged strongly. Today, its service sector is the fastest-growing sector in the world, contributing to more than 60% to its economy and accounting for 28% of employment. Manufacturing remains as one of its crucial sectors and is being given due push via the governments' initiatives such as "Make in India". Although the contribution of its agricultural sector has declined to around 17%, it still is way higher in comparison to the western nations. The economy's strength lies in a limited dependence on exports, high saving rates, favorable demographics, and a rising middle class [2].

PwC's analysis of key sectors such as education, healthcare, agriculture, financial services, power, manufacturing, retail, urbanization, digital and physical connectivity suggests that new solutions are necessary in each sector. As the world increasingly confronts technological change and sustainability challenges, we believe India and the Winning Leap can offer an exemplar for other growth markets [3].

The data for this study report is taken from the World Bank open data source, with 319 variables comprising of indicators of health, education, productivity, consumption spending, investment spending, import, exports, government spending, etc. and varying over the years 1960 to 2017 (58 years). A comprehensive study of the various indicator variables like Service value added, Industrial contribution, Manufacturing value added, agriculture value added, merchandise exports, number of infant deaths, fuel imports, urban rural population count, inflation, government consumption etc. gave an insight as to which variables influence the calculation of Economic growth of a country and how over the years (changes due to government policies) has evolved the GDP. The model developed from this study is used to for prediction of GDP of India.

2. INTRODUCTION

a. Importance

Importance of GDP lies in the fact that a healthy GDP reflects how well is the country growing on a global platform. It is usually the factor of comparison between economies across the globe. When GDP is growing quickly, you often see more jobs, rising wages, and better profits for businesses [4].

Samuelson and Nordhaus neatly sum up the importance of the national accounts and GDP in their seminal textbook "Economics." They liken the ability of GDP to give an overall picture of the state of the economy to that of a satellite in space that can survey the weather across an entire continent. GDP enables policymakers and central banks to judge whether the economy is contracting or expanding, whether it needs a boost or restraint, and if a threat such as a recession or inflation looms on the horizon [5].

The resulting GDP is an integration of consumption spending by households, investment spending by businesses and households, government purchases of goods and services, net exports or net foreign demand, and the literacy rate of the economy. Literacy rate indirectly affects the GDP since it churns the skilled workforce into the economy reflecting the monetary realization of businesses, which in turn drives the economy [4].

Since GDP is one of the primary indicators used to gauge the health of a country's economy. At present, India has the fastest growing GDP rate in the world. As one of the developing countries, analysis of its socio-economic data over the years could yield great insights for setting up a path for other developing or under-developed countries which have similar per capita income, share the same history and culture, have similar ethnic tension and conflict, etc. An analysis for the success over GDP growth for a developing nation like India, is therefore note-worthy [6].

b. Objective

Objective of this report is to perform a predictive assessment on the GDP (Gross Domestic Product) from the World Bank dataset, India through an inferential analysis of various socio-economic factors. Towards achieving the above objective, we have implemented the methods learned in ISEN-613 Engineering Data Analysis on a dataset having more than 100 variables.

c. Scope

The scope of this project is to identify the socio-economic factors that affect the GDP rate of India, in general for countries rapidly transitioning to developed economies. While nearly 190 predictors were analyzed for their impact, more data is needed for future predictions of GDP over the years. This project presents a broad view of various socio-economic factors that affect GDP in general and can be used as a base for a comparative study for GDP calculation for similar countries.

3. APPROACH

The dataset we have used is obtained from The World Bank Library and includes several (319) shortlisted to 190 socio-economic factors that can used to understand the GDP rate of India from 1960-2017.

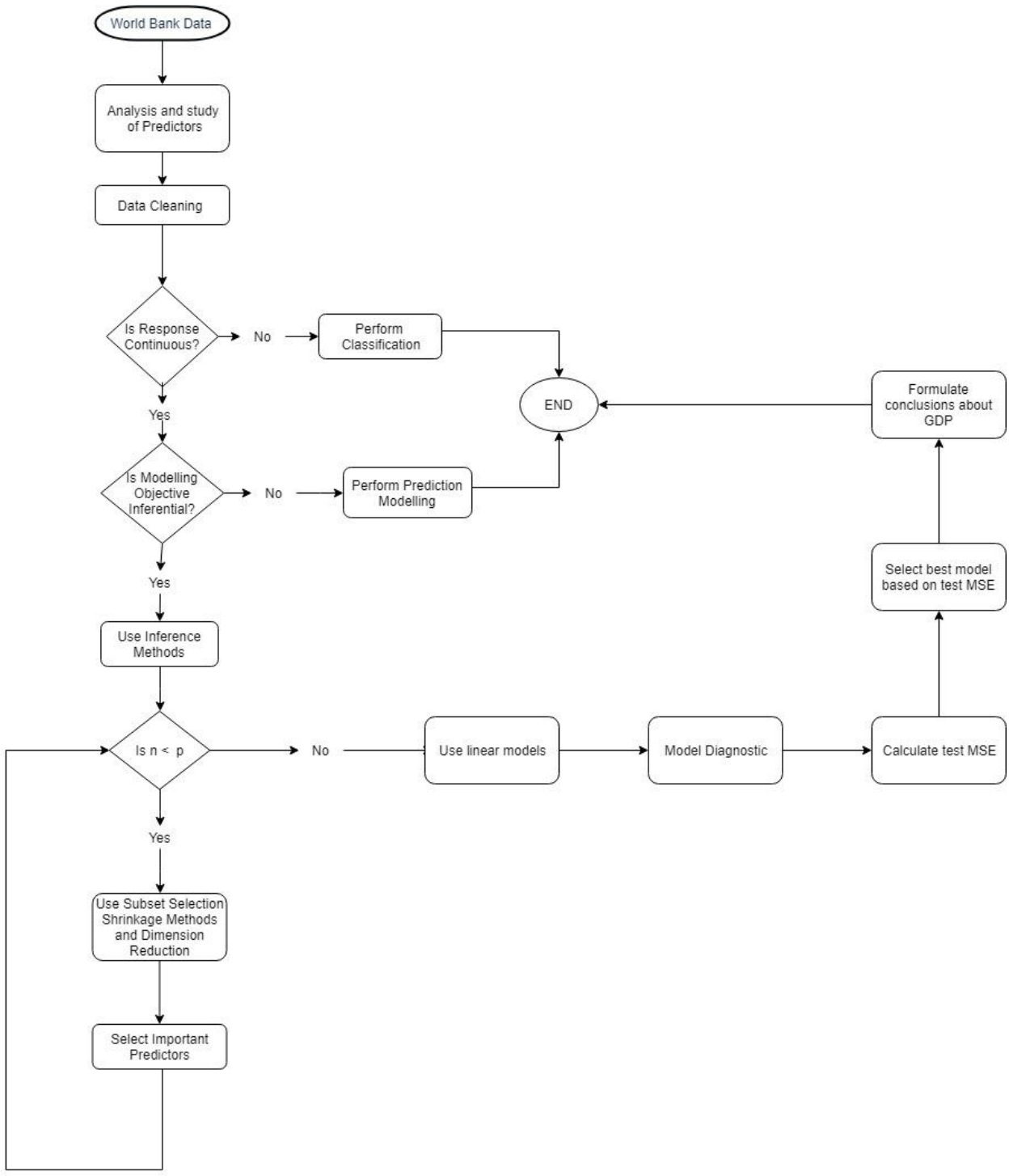

Figure 0: Flow chart for data analysis

Step 1: The dataset comprises of 319 predictors, most of which have no value recorded. Since the missing values in the dataset cannot be imputed with accuracy given the small number of observations (n=58) recorded and would unfairly impact the results of our prediction. We thus, choose to remove 129 predictors from our data which have more than 50 (85%) missing observations.

Step 2: Next, we studied, the physical meaning of each of the predictors in the final dataset with shortlisted 190 predictors (p) and 58 observations(n) obtained in step 1.

Step 3: On this final dataset we performed exploratory data analysis (EDA), calculating statistical parameters like max, min, median and mean. This gave us a better understanding to conclude that the data is not standardized. Due to the difference in scale of various predictors, their coefficients in the model would turn out to be biased, hence we chose to standardize the give dataset.

Step 4: Apart from standardization, correlation plays an important part in data analysis. Based from the value in the correlation matrix, the variables with high correlation coefficients (multicollinearity) are addressed through model diagnosis techniques.

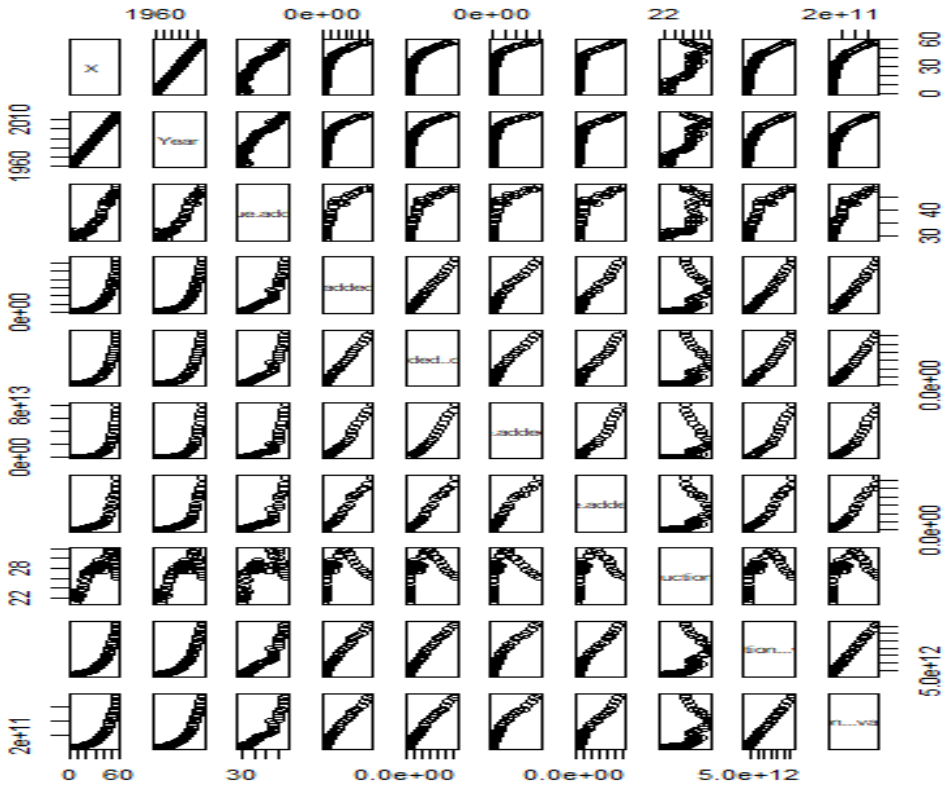

Figure 1: Graph depicting correlation of some predictors

Step 5: Starting with the model building, since the response variable (GDP) is numerical we choose to proceed using linear regression statistical learning method. Also, our dataset (58 X 190) has n<p Ordinary Least Squares (OLS) parameter estimation methods cannot be applied since the variance of the parameter estimate(s) would be infinite. We thus choose Advanced Linear Regression Methods like subsets selection, shrinkage, regression trees and PCA for our prediction.

Reasons:

Table 1: Statistical Learning methods considered

Subset Selection	Best Subset Selection	✘	Computationally intensive with 2^{190} models, not feasible
	Backward Selection	✘	n<p, it cannot be pursued since it chooses the full model to start with variable selection
	Forward Selection	✓	Possible. Number of models 1+ 190(190 +1)/2 = 18146
Shrinkage Methods	Ridge Regression	✓	Reduces variance, but does not improve interpretability
	Lasso	✓	Feature Selection can be done as it reduces the coefficients of insignificant predictors to exactly zero.
Regression Trees		✓	For increasing model interpretability …depends on linear or non-linear relationship between predictors and response. Accuracy of tress can be improved by the following methods, 1. Bagging 2. Boosting 3. Random Forests
PCA		✓	To retain maximum information of the correlated variables and to reduce the dimension of the dataset.

Step 6: After model fitting in step 5, we check for the model accuracy based on 5 potential problems through model diagnosis techniques; non-linearity of variables, non-constant variance (transform the response) of error terms, high leverage points, outliers and collinearity (variance inflation factor > 5) of predictors. The points identified through the model diagnosis were then treated reasonably so that they do not affect the resulting model.

Step 7: The optimal models obtained from each method were tested based on the value of test MSE to select the best model. We do this by fitting our model on the training dataset and predicting on the test dataset. (methods like: LOOCV, K-fold)

Step 8: We then select the model which gives the least value for test MSE.

4. IMPLEMENTATION

a. Subset Selection Methods

The subset selection methods offer 3 methods to choose from, namely, Forward Selection, Backward selection and the Best Subset Selection. The Best Subset selection, though being a very accurate method requires to run 2^p i.e. 2^{190} models which would be highly computationally intensive. We thus choose to implement other methods.

The backward selection method requires the full model to start with and progressively removes the variables with each run. Our dataset however has n<p thus invalidating the use of backward selection as an approach.

The forward selection method offers a great way to select significant predictors in the model by not being too computationally intensive as well as works well for datasets with n<p. Hence, we select this method for our analysis.

i. Forward Subset Selection

Forward Subset Selection method indicated only 30 predictors in the best model, which was selected based on indirectly estimating the test error by making adjustment to the training error to account for the bias of overfitting as number of predictors increase. The BIC and R-adjusted show similar for minimum error, resulting in the best model comprising of 30 variables. We choose the model with 30 predictors and carry out a multiple linear regression on the reduced number of predictors.

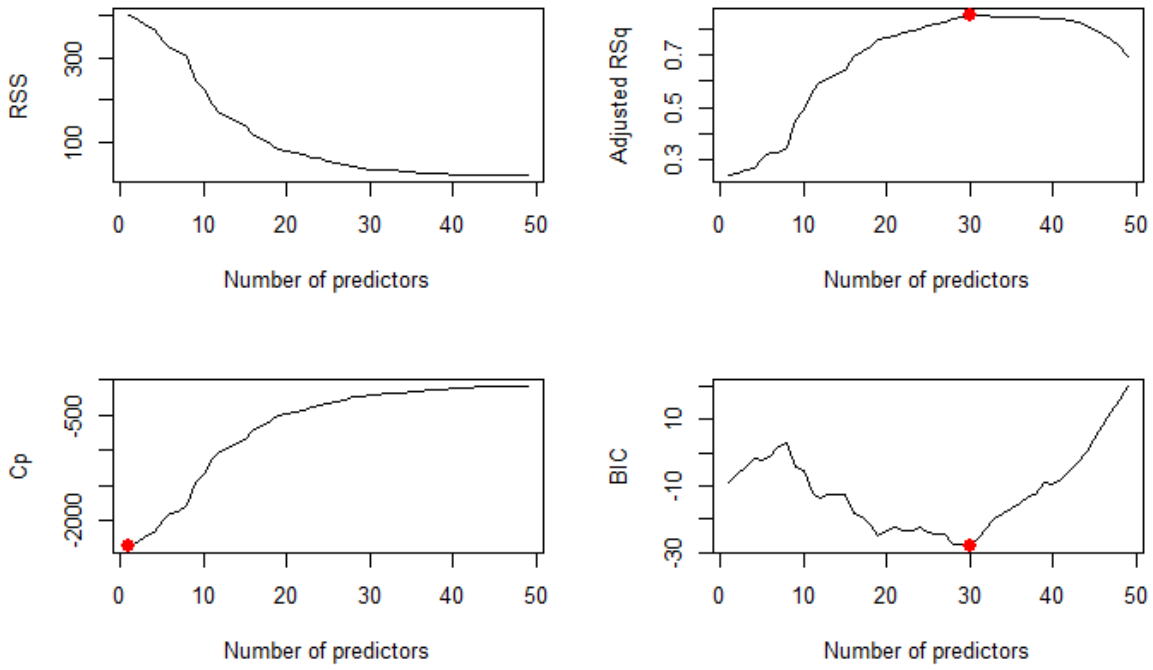

Figure 2: Indirect estimator of test error- RSS, Adj R^2, Cp, BIC

After fitting the regression model using the predictors from Forward Stepwise Selection, we assessed the model accuracy by checking for the five potential problems: high-leverage points, collinearity, non-constant variance of error terms, non-linearity and outliers.

1. High Leverage Points: We chose to eliminate points with Cook's distance more than 33%. In the diagnosis however, there were no points identified as high leverage points.
2. Outliers: Observing the diagnostic plots the points 6, 28, 36 came across as potential outliers and removed them from the model.
3. Multi-collinearity: Observations with variance inflation factor (VIF) > 5, were removed in sequential steps leaving us with 9 observations in the final model with VIF <5.

4. Non-constant variance of error terms: The assumption of the linear model being that the error terms have a constant variance. We thus look for points with a non-constant error term. We removed all such points which had a p-value less than 0.01.

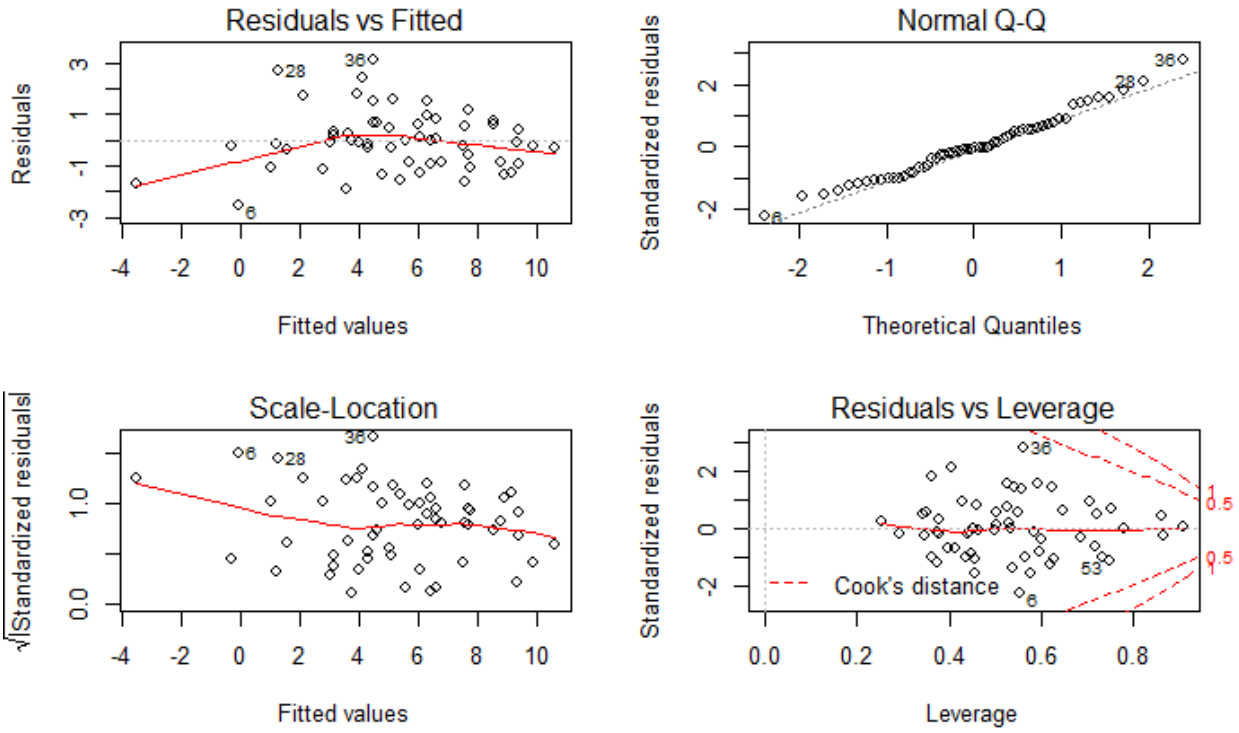

Figure 3: Model Diagnostic plots

The final model obtained after model diagnosis comprised of predictors and their co-efficient estimates as shown in figure 4.

```
Population Ages- 40-44 (Male)
Population Ages- 35-39 (Female)
Population Ages- 25-29 (Female)
Population Ages- 15-64
Consumer Price Index
Discrepency in Expenditure estimate of GDP
Gross National Expenditure
Military Expenditure
Arms Imports
```

```
Call:
lm(formula = (GDP.growth) ~ `64` + `67` + `71` + `74` + `103` +
    `126` + `165` + `187` + `190`, data = Data.train)

Residuals:
    Min      1Q  Median      3Q     Max
-4.7747 -1.1866 -0.3253  1.2690  4.6871

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.949e+00  4.069e+01  -0.072    0.943
`64`         1.988e+00  1.714e+00   1.160    0.255
`67`        -5.305e-01  1.499e+00  -0.354    0.726
`71`         2.538e+00  1.683e+00   1.508    0.141
`74`        -8.169e-01  1.621e+00  -0.504    0.618
`103`        6.625e-02  9.391e-02   0.705    0.485
`126`       -5.478e-13  5.356e-13  -1.023    0.314
`165`       -9.095e-02  4.214e-01  -0.216    0.830
`187`        3.706e-02  1.000e+00   0.037    0.971
`190`       -4.941e-10  4.835e-10  -1.022    0.314

Residual standard error: 2.242 on 33 degrees of freedom
Multiple R-squared:  0.3812,    Adjusted R-squared:  0.2125
F-statistic: 2.259 on 9 and 33 DF,  p-value: 0.04263
```

Figure 4: Best model obtained from Subset selection methods

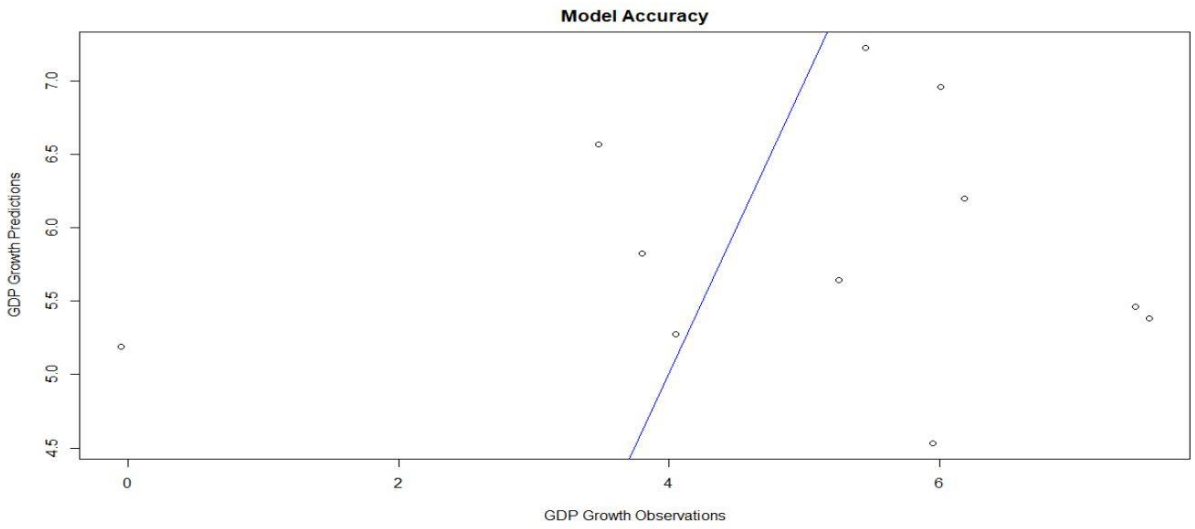

Figure 5: Model Accuracy for MLR on best subset by Forward Subset Selection

b. Shrinkage methods

Shrinkage methods, fits a model containing all p predictors by shrinking the coefficients of insignificant terms towards zero. The idea behind this method is to improve prediction performance by reducing the variance with the creation of a more interpretable model containing less parameters. This method is particularly important for the large number of predictors as in our data. In our analysis, we have tried both shrinkage methods, Ridge and LASSO.

i. Ridge Regression

When the data has $n<p$, then there is no longer a unique solution to the ordinary least squares coefficient estimates because the variance becomes infinite. Ridge regression regularizes the coefficient estimates with the use of a shrinkage parameter λ (lambda). We implemented Ridge Regression in R, and then split the data into train and test to perform cross validation for the best model at the best lambda value. From this procedure, the optimal model had a lambda value of 52.13, based on the lowest test MSE.

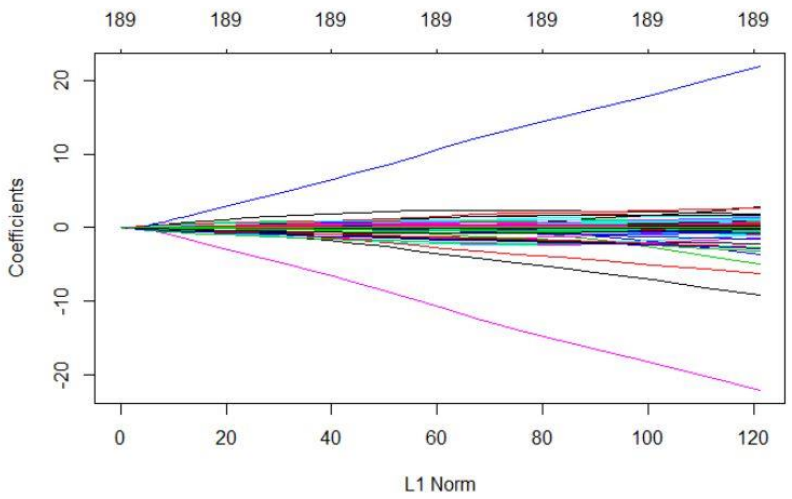

Figure 6: Standardized ridge regression coefficients as a function of L1 norm

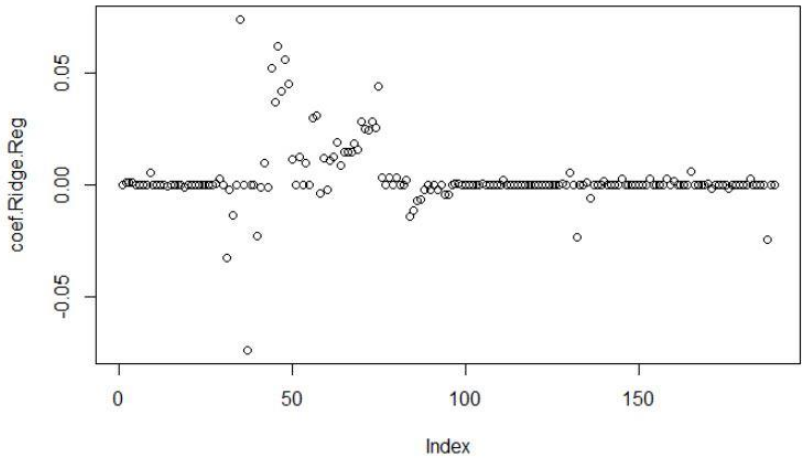

Figure 7: Ridge Regression coefficient estimates

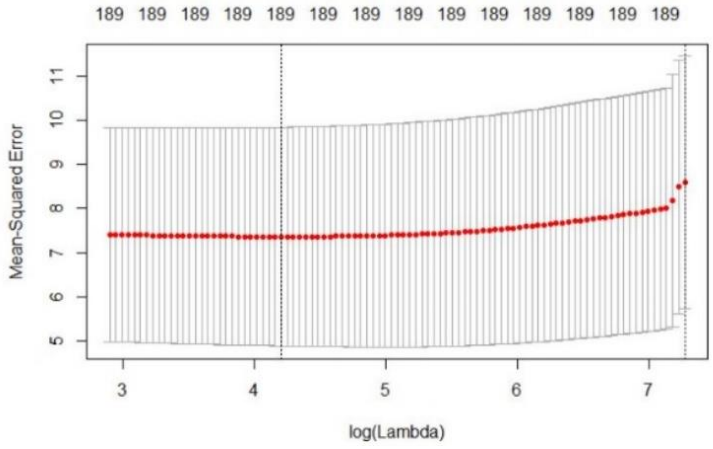

Figure 8: Cross-validation errors at different values of lambda

Since ridge regression doesn't entirely make the coefficients of insignificant variables zero, we choose to put a threshold of 10^{-3} on the estimate of coefficients, i.e. any value below 0.001 was termed as zero and such predictors were ignored from the model accordingly.

The final model gave a test MSE of 10.51.

ii. Lasso

Lasso is another shrinkage method selected for our model. Since the number of predictors are more than the number of observations (n<p), in our dataset shrinkage of coefficients of insignificant predictors is an important approach. Lasso shrinks the coefficients for such predictors to exactly zero in contrast to the Ridge regression method using a shrinkage parameter, λ (lambda) to facilitate this. Therefore, the output of LASSO yields a model as a subset of the original variables.

Lasso gives good results in the case when there are a small number of predictors to consider with much difference in coefficients, such that the ones with insignificant values can be shrunk to zero. It thus improves the model accuracy by variable selection method, and hence interpretability of the model as well.

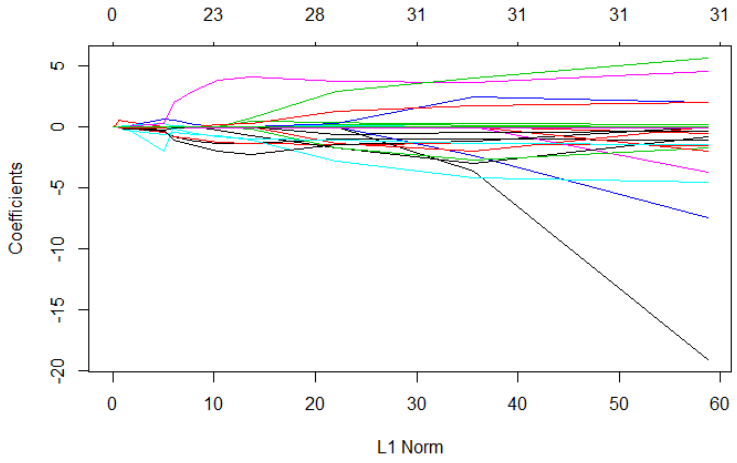

Figure 9: LASSO Coefficient Shrinking

After implementing the LASSO in R, the best model gives a lambda value of 0.4753534 (based on lowest MSE). The cross-validated plot shows the model has 5 variables, as can be seen in the figure 10 below.

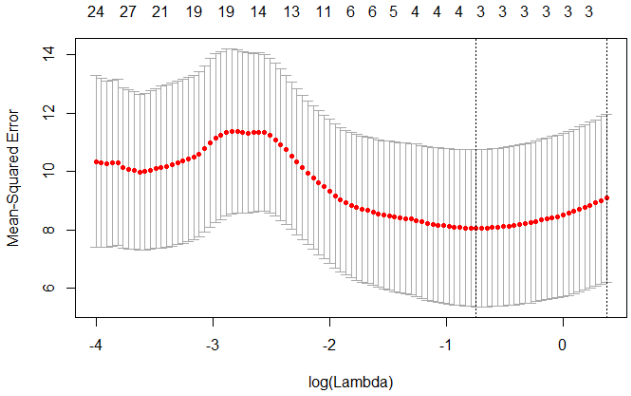

Figure 10: MSE for Lasso at values of lambda

Thereafter fitting a linear model to the predictors selected through LASSO, we achieved the following results of the intercept and coefficient values for each predictor:

Linear model:

lm(formula = GDP.growth ~ `X69` + `X110` + `X127` + `X150`, data = Data)

Coefficients:

Table 2:Coefficient estimates table, Lasso

	Coefficient	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-	-5.525e+00	7.480e+00	-0.739	0.463
X69	Population age 25-29, male	1.354e+00	1.220e+00	1.110	0.272
X110	Gross Domestic savings	4.347e-02	8.970e-02	0.485	0.630
X127	GDP deflator	-6.651e-13	4.639e-13	-1.434	0.157
X150	Changes in inventories	3.598e-13	3.619e-13	0.994	0.325

The fitted model had the following characteristics:

Residual standard error: 2.662 on 53 degrees of freedom
Multiple R-squared: 0.2982, Adjusted R-squared: 0.2452
F-statistic: 5.63 on 4 and 53 DF, p-value: 0.0007485

Further model diagnostics was performed on the fitted linear model on the predictors shortlisted by Lasso, to identify and remove high leverage points, collinear predictor variables using variance inflation factor (VIF), confirm linearity of response vs variables, non-constant variance of error terms (heteroscedasticity), and identify outliers.

1. High leverage points: The cook's distance was used to filter out the high leverage points, points with distance greater than 33% were removed.
2. Collinearity: significant collinearity between the predictor variables exists, to identify and remove the collinear variables we relied on the variance inflation factor (VIF). An iterative approach was followed to remove closely related predictor variables with VIF >5 from the model. Variable 'X110'- Gross Domestic savings was removed from the model.
3. Non-linearity: Based on the residual plots, we can see that the relationship between the predictors and response is linear.
4. Non-constant variance of error terms or heteroscedasticity: The absence of a funnel shape in the residual plot indicates that our assumption of having a constant variance of error terms holds true.
5. Outliers: As can be seen from residual plots only point 20 is an outlier, since it cannot be confirmed as an error of data collection or otherwise, we cannot remove it from our analysis.

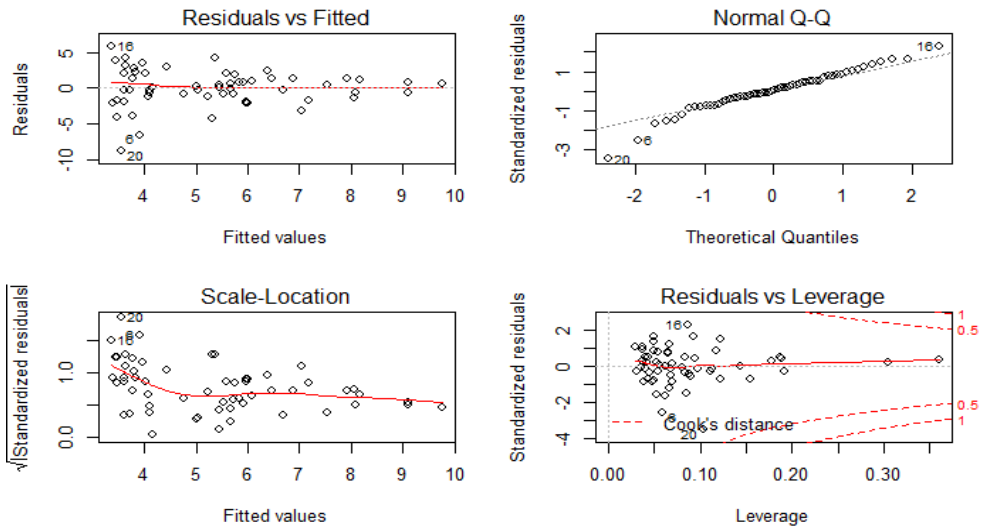

Figure 11: Model Diagnostic plot for LASSO

After the model diagnostics, we fitted the linear model again using the dataset with 58 observations (without the high leverage points and using only the non-collinear predictor variables, 3). We got the following model:

lm(formula = GDP.growth ~ `X69` + `X127` + `X150`, data = Data)

Coefficients:

Table 3: Coefficient estimates of Lasso (2)

	Coefficient	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-	-7.474e+00	6.261e+00	-1.194	0.2378
X69	Population age 25-29, male	1.736e+00	9.240e-01	1.879	0.0657
X127	GDP deflator	-7.243e-13	4.444e-13	1.630	-0.1089
X150	Changes in inventories	4.624e-13	2.914e-13	1.587	0.1184

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

The fitted final model had the following characteristics:

Residual standard error: 2.643 on 54 degrees of freedom
Multiple R-squared: 0.2951, Adjusted R-squared: 0.2559
F-statistic: 7.535 on 3 and 54 DF, p-value: 0.0002668

Calculated Test MSE from this model is 6.847214.

The test MSE via CV- k-fold (10) approach on the same model gives a value of 7.313129.

c. Tree-Based Methods

Regression trees divides the predictor space into different regions and graphically represents them with a tree structure, which makes them very easy to understand. However, we see in our models so far that the relationship between the different predictors and the response is almost linear, thus the other linear regression models will perform better than regression trees. We however chose to run the random forest model on our dataset to check if that was untrue.

i. Random Forest

Since the predictors in our dataset are highly correlated we skipped directly to random forest which generates trees after decorrelating them, which improves the performance of the model by reducing the variance. It only considers m (m<p) predictors at each split of the tree to avoid similarity of models by avoiding the usage of the same m important predictors in the model each time. Since we are using Random Forests for regression, m=p/3 (63) was used in as the function. Figure 12, below shows the resulting importance of the variables.

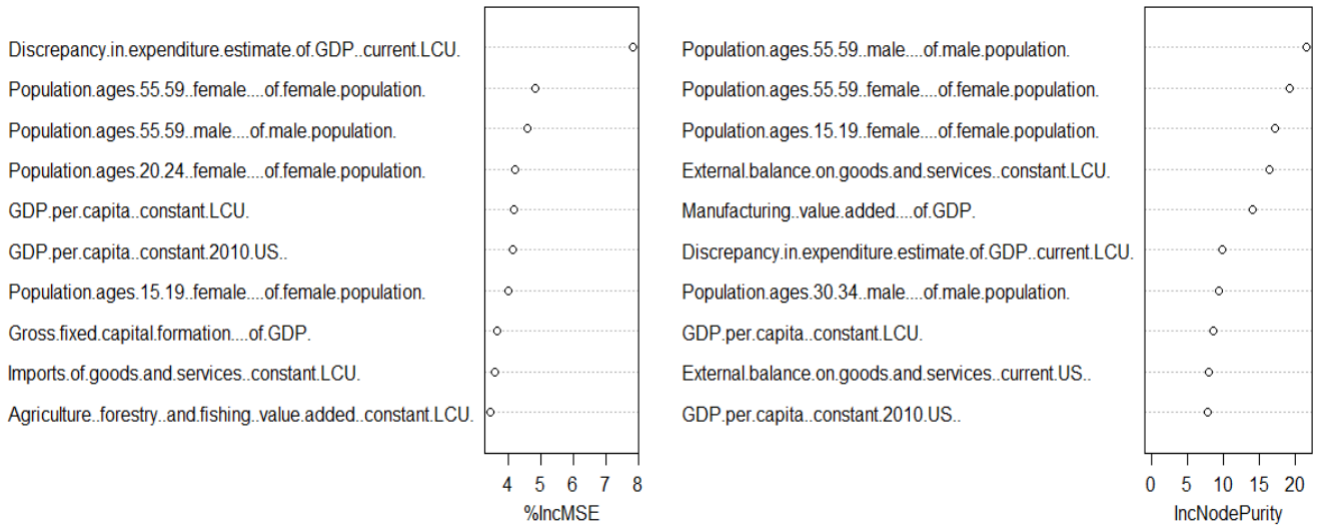

Figure 12: Importance of variables based on MSE and Node Purity

The figure below shows the decrease in error as the number of trees increases, at m=63.

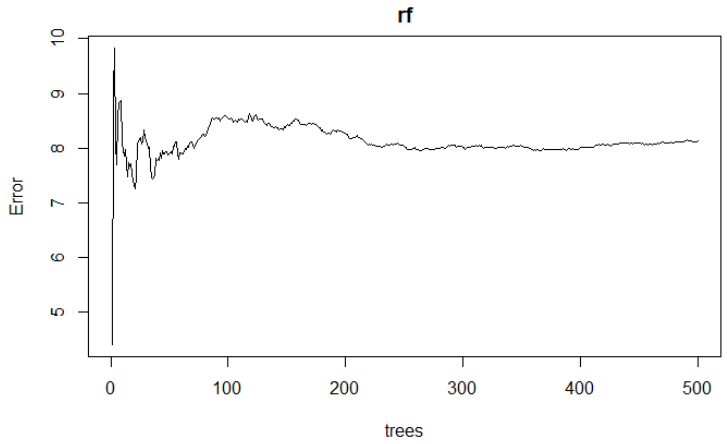

Figure 13: Error as a function of Number of trees

The final model from random forest gives a test MSE of 15.64348.

d. Principal Component Analysis:

Curse of dimensionality is something that needs to be dealt with while working in multivariate environment. The principal component analysis transforms the data into mutually orthogonal and uncorrelated principal components which are used to analyze the data. Hence PCA is reasonable thing to do while working with many predictors.

Since our dataset contained many predictors, principal component analysis was one of the methods utilized to reduce the dimension of our model. When we checked the means and variances of each predictor, there are large differences between the predictors. Because of this, we standardized the variables to have zero mean and standard deviation of one before performing PCA. The predictors that PC1 and PC2 explains is in Figure 14 below.

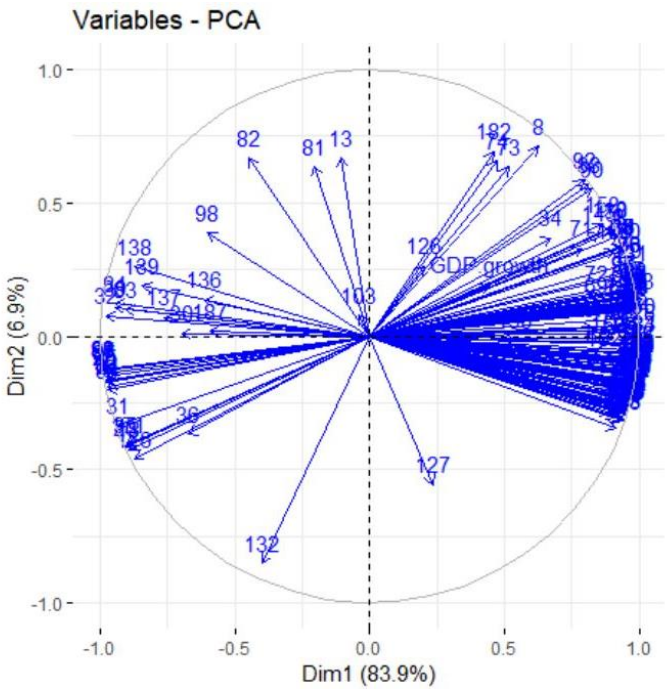

Figure 14: Biplot of Principal Component Scores and Principal Component Loadings of PC1 and PC2.

The results of PCA shows that the first Principal Component 1 explains 84.21% of the variance and the Principal Component 2 explains about 6.9%, with decreasing variance explanation. By examining the scree plot in Figure 14 below, the elbow is at principal component 2. Components 3 and higher each explains 2% and less, making them essentially worthless in our analysis.

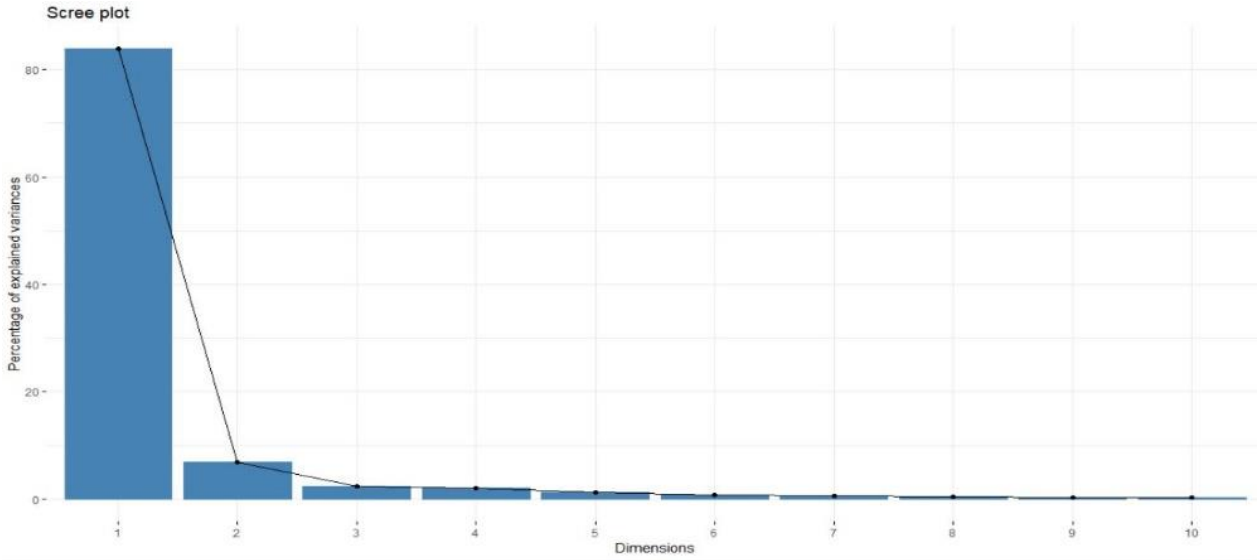

Figure 15: Scree Plot, Proportion of Variance Explained by each Principal Component.

We conducted PCR analysis on the model. The test MSE obtained was 9.2712. The CV errors of individual Principal components and the variance explained by each of them of the final model are as depicted in the figure 16 below.

```
Data:   X dimension: 58 188
        Y dimension: 58 1
Fit method: svdpc
Number of components considered: 51

VALIDATION: RMSEP
Cross-validated using 10 random segments.
       (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
CV           3.091    2.828    2.816    2.867    2.865    2.896    3.060    3.068
adjCV        3.091    2.822    2.809    2.858    2.853    2.883    3.038    3.044
       8 comps  9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps
CV       3.087    3.101     3.354     3.332     3.342     3.453     3.479     3.484
adjCV    3.059    3.074     3.313     3.283     3.300     3.407     3.433     3.448
       16 comps  17 comps  18 comps  19 comps  20 comps  21 comps  22 comps  23 comps
CV        3.377     3.640     3.582     3.747     3.840     3.894     3.684     3.538
adjCV     3.324     3.587     3.516     3.675     3.759     3.802     3.570     3.446
       24 comps  25 comps  26 comps  27 comps  28 comps  29 comps  30 comps  31 comps
CV        3.572     3.635     3.832     4.091     3.847     4.140     4.046     3.793
adjCV     3.484     3.562     3.746     3.988     3.734     4.033     3.956     3.653
       32 comps  33 comps  34 comps  35 comps  36 comps  37 comps  38 comps  39 comps
CV        3.393     3.037     2.693     2.567     2.778     2.780     2.765     2.698
adjCV     3.272     2.944     2.598     2.488     2.659     2.669     2.657     2.599
       40 comps  41 comps  42 comps  43 comps  44 comps  45 comps  46 comps  47 comps
CV        2.894     3.431     4.370     5.307     5.105     6.904     6.620     7.132
adjCV     2.787     3.289     4.176     5.058     4.870     6.570     6.303     6.792
       48 comps  49 comps  50 comps  51 comps
CV        7.114     7.293     7.223     6.624
adjCV     6.777     6.934     6.865     6.285

TRAINING: % variance explained
            1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
X             84.21    91.11    93.49    95.49    96.70    97.52    98.09    98.53
GDP.growth    19.38    22.73    23.36    24.76    24.78    24.93    25.90    27.57
            9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps  16 comps
X             98.84     99.13     99.31     99.48     99.59     99.68     99.74     99.79
GDP.growth    27.72     28.11     30.87     30.92     31.16     31.43     32.55     38.93
            17 comps  18 comps  19 comps  20 comps  21 comps  22 comps  23 comps
X              99.84     99.87     99.90     99.91     99.93     99.94     99.95
GDP.growth     41.29     45.80     46.21     47.48     51.72     59.29     60.60
            24 comps  25 comps  26 comps  27 comps  28 comps  29 comps  30 comps
X              99.96     99.97     99.97     99.98     99.98     99.99     99.99
GDP.growth     61.65     61.67     63.28     65.54     69.84     69.85     71.79
            31 comps  32 comps  33 comps  34 comps  35 comps  36 comps  37 comps
X              99.99     99.99     99.99    100.00    100.00    100.00    100.00
GDP.growth     81.79     84.81     86.76     89.86     90.03     92.05     92.05
            38 comps  39 comps  40 comps  41 comps  42 comps  43 comps  44 comps
X             100.00    100.00    100.00    100.00     100.0     100.0    100.00
GDP.growth     92.07     92.08     92.08     92.29      92.3      92.5     92.56
            45 comps  46 comps  47 comps  48 comps  49 comps  50 comps  51 comps
X             100.00    100.00    100.00    100.00       100    100.00    100.00
GDP.growth     92.88     93.19     93.24     93.83        96     96.55     98.65
```

Figure 16: Summary of results obtained from PCR

5. MODEL COMPARISON

While implementing the data analysis models on the given data we have observed that there was a very steep decline in GDP growth in1979 from 5.7% to -5.2 %. To investigate this abnormality we studied the historical events that occurred during these couple of years to make sure that this point was not an outlier. The circumstances that caused this were severe drought, abrupt decline in average rainfall leading to agricultural production decline by 10%, poor industrial growth, high inflation and rise in crude oil prices. As India is heavily dependent on rain for its agricultural output and she imports most of her oil from Gulf countries, this was the most probable outcome and hence the steep decline.

The model performance comparison is shown below:

Model	Test MSE
Forward Subset Selection Method	5.228747
Ridge Regression	10.51
LASSO Regression	6.847214
Random Forest	15.64348
Principal Component Regression	9.2712

6. CONCLUSION

The team learned a lot in the data analysis using statistical learning methods domain through the project. The major points can be summarized as follows:

- We first carried out Subset Selection wherein we chose forward subset selection method and obtained the test MSE as 5.228747.
- We then conducted Shrinkage methods, Ridge and Lasso obtaining test MSEs, 10.51 and 6.847214 respectively.
- Tree based methods like Random Forests were used to produce interpretable, reduced variance trees. The test MSE using random forests was 15.64348.
- Later, to reduce the huge number of predictors, we conducted Principal Components Regression and the test MSE obtained was 9.2712
- Of the models evaluated, we found Forward subset selection method fits our model best with test MSE, 5.228747. The best model hence is given in Equation below.

GDP Growth = – 2.949 + 1.988 *(Population age 40-44, male)
– 5.305e-01 * (Population age 35-39, female)
+ 2.538 * (Population age 25-29, female)
– 8.169e-01 * (Population age 15-64, Total)
– 6.625e-02 * (Consumer Price Index)
– 5.478e-13 * (Discrepancy in Expenditure Estimate of GDP)
– 9.095e-02 * (Gross National Expenditure)
+ 3.706e-02 * (Military Expenditure)
– 4.941e-10 * (Arms Import)

- As we know that GDP of any country mainly depends on the functionality of the government and the population output of the country. In case of India the revenue generated by the informal economy contributes heavily as far as GDP is concerned. Our data does not contain this aspect hence this fact may lead to error in the projection of GDP. Our model also shows that the working population of the nation, Government Spending and Consumer Price Index and our model depicts the same.
- Based on our project work, the future models can contain better predictors from economic point of view that give insights into country's agricultural data, banking data and demographics. We believe that by including these kind of predictors, we can produce better accuracy in implementing the models.

7. REFERENCES

[1] https://ourworldindata.org/economic-growth

[2] https://www.investopedia.com/insights/worlds-top-economies/

[3] https://www.pwc.in/assets/pdfs/future-of-india/future-of-india-the-winning-leap.pdf

[4] https://www.quora.com/What-is-GDP-and-why-is-it-so-important

[5] https://www.investopedia.com/articles/investing/121213/gdp-and-its-importance.asp

[6] https://www.hindustantimes.com/india-news/india-at-70-a-comparison-of-progress-with-pakistan-and-4-other-nations-here/story-UEIuCPfvjOFE4U3aFHaIYI.html

[7] Gareth James, Danielia Witten, Trevor Hastie, Robert Tibshirani, book titled "An introduction to Statistical Learning with applications in R", Springer, 2015

[8] https://www.ceicdata.com/en/indicator/india/gdp-per-capita

[9] https://tradingeconomics.com/india/gdp-per-capita

8. APPENDIX

```
library(ggplot2)
library(randomForest)
library(MASS)
library(xgboost)
library(glmnet)
library(neuralnet)
library(VIM)
library(mice)
library(leaps)

data<-read.csv(file.choose(),header=T)

sum(is.na(data))

missingdata<-as.data.frame(sort(sapply(data, function(x) sum(is.na(x))),decreasing=T))
missingdata<-(missingdata/nrow(data))*100
class(missingdata)
colnames(missingdata)[1]<-"Percentage"
missingdata$predictors<-row.names(missingdata)
missingdata

x<-subset(missingdata,missingdata$Percentage==0)

Data<-data[,(x$predictors)]

for (i in 1:190)
{
  names(Data)[i]<-paste(i)
}

names(Data)[118]<-paste("GDP.growth")

Data

lm.fwd.fit0=lm(GDP.growth~.,data=Data)
sum<-summary(lm.fwd.fit0)

regfit.fwd=regsubsets(GDP.growth~.,data=Data,nvmax=189, method="forward")
summ<-summary(regfit.fwd)

par(mfrow=c(2,2))
plot(summ$rss,xlab="Number of predictors",ylab="RSS",type="l")

plot(summ$adjr2,xlab="Number of predictors",ylab="Adjusted RSq",type="l")
which.max(summ$adjr2)
points(30,summ$adjr2[30], col="red",cex=2,pch=20)

plot(summ$cp,xlab="Number of predictors",ylab="Cp",type='l')
which.min(summ$cp)
points(1,summ$cp[1],col="red",cex=2,pch=20)

plot(summ$bic,xlab="Number of predictors",ylab="BIC",type='l')
which.min(summ$bic)
points(30,summ$bic[30],col="red",cex=2,pch=20)
```

```
coef(regfit.fwd,30)

lm.fwd.fit1=lm(GDP.growth~`4`+`14`+`19`+`21`+`23`+`27`+`28`+`43`+`64`+`65`+`66`+`67`+`71`+`74`+`93`+`94`+`103`+`110`+`126`+`130`+`140`+`144`+`150`+ `151`  +`153`+`165`+`186`+`187`+`189`+`190`,data=Data)
summary(lm.fwd.fit1)
plot(lm.fwd.fit1)

library(stats)
hlp=cooks.distance(lm.fwd.fit1)>0.333
hlp

library(car)
vif(lm.fwd.fit1)
max(vif(lm.fwd.fit1))

##Remove '28'

lm.fwd.fit1.whlp2=lm(GDP.growth~`4`+`14`+`19`+`21`+`23`+`27`+`43`+`64`+`65`+`66`+`67`+`71`+`74`+`93`+`94`+`103`+`110`+`126`+`130`+`140`+`144`+`150`+ `151`  +`153`+`165`+`186`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp2)
max(vif(lm.fwd.fit1.whlp2))

##Remove '94'

lm.fwd.fit1.whlp3=lm(GDP.growth~`4`+`14`+`19`+`21`+`23`+`27`+`43`+`64`+`65`+`66`+`67`+`71`+`74`+`93`+`103`+`110`+`126`+`130`+`140`+`144`+`150`+ `151`  +`153`+`165`+`186`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp3)
max(vif(lm.fwd.fit1.whlp3))

##remove '130'

lm.fwd.fit1.whlp4=lm(GDP.growth~`4`+`14`+`19`+`21`+`23`+`27`+`43`+`64`+`65`+`66`+`67`+`71`+`74`+`93`+`103`+`110`+`126`+`140`+`144`+`150`+ `151`  +`153`+`165`+`186`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp4)
max(vif(lm.fwd.fit1.whlp4))

##Remove '4'

lm.fwd.fit1.whlp5=lm(GDP.growth~`14`+`19`+`21`+`23`+`27`+`43`+`64`+`65`+`66`+`67`+`71`+`74`+`93`+`103`+`110`+`126`+`140`+`144`+`150`+ `151`  +`153`+`165`+`186`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp5)
max(vif(lm.fwd.fit1.whlp5))

##Remove '93'

lm.fwd.fit1.whlp6=lm(GDP.growth~`14`+`19`+`21`+`23`+`27`+`43`+`64`+`65`+`66`+`67`+`71`+`74`+`103`+`110`+`126`+`140`+`144`+`150`+ `151`  +`153`+`165`+`186`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp6)
max(vif(lm.fwd.fit1.whlp6))

##Remove '14' & '21'
```

```
lm.fwd.fit1.whlp7=lm(GDP.growth~`19`+`23`+`27`+`43`+`64`+`65`+`66`+`67`+`71`+`74`+`103`+`110`+`126`+`
140`+`144`+`150`+ `151`  +`153`+`165`+`186`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp7)
max(vif(lm.fwd.fit1.whlp7))

##Remove '186'& '140'

lm.fwd.fit1.whlp8=lm(GDP.growth~`19`+`23`+`27`+`43`+`64`+`65`+`66`+`67`+`71`+`74`+`103`+`110`+`126`+`
144`+`150`+ `151`  +`153`+`165`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp8)
max(vif(lm.fwd.fit1.whlp8))

##Remove '66' '144'

lm.fwd.fit1.whlp9=lm(GDP.growth~`19`+`23`+`27`+`43`+`64`+`65`+`67`+`71`+`74`+`103`+`110`+`126`+`150`+
`151`  +`153`+`165`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp9)
max(vif(lm.fwd.fit1.whlp9))

##Remove '19' '23'

lm.fwd.fit1.whlp10=lm(GDP.growth~`27`+`43`+`64`+`65`+`67`+`71`+`74`+`103`+`110`+`126`+`150`+ `151`
+`153`+`165`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp10)
max(vif(lm.fwd.fit1.whlp10))

##Remove '43'

lm.fwd.fit1.whlp11=lm(GDP.growth~`27`+`64`+`65`+`67`+`71`+`74`+`103`+`110`+`126`+`150`+ `151`
+`153`+`165`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp11)
max(vif(lm.fwd.fit1.whlp11))

##Remove '150' '110'

lm.fwd.fit1.whlp12=lm(GDP.growth~`27`+`64`+`65`+`67`+`71`+`74`+`103`+`126`+ `151`
+`153`+`165`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp12)
max(vif(lm.fwd.fit1.whlp12))

##Remove '65'

lm.fwd.fit1.whlp13=lm(GDP.growth~`27`+`64`+`67`+`71`+`74`+`103`+`126`+ `151`
+`153`+`165`+`187`+`189`+`190`,data=Data)

vif(lm.fwd.fit1.whlp13)
max(vif(lm.fwd.fit1.whlp13))

##Remove '189' '27'
```

```
lm.fwd.fit1.whlp14=lm(GDP.growth~`64`+`67`+`71`+`74`+`103`+`126`+ `151`
+`153`+`165`+`187`+`190`,data=Data)

vif(lm.fwd.fit1.whlp14)
max(vif(lm.fwd.fit1.whlp14))

##Remove '153' '151'

lm.fwd.fit1.whlp15=lm(GDP.growth~`64`+`67`+`71`+`74`+`103`+`126`+`165`+`187`+`190`,data=Data)

vif(lm.fwd.fit1.whlp15)
max(vif(lm.fwd.fit1.whlp15))

ncvTest(lm.fwd.fit1.whlp15)
plot(lm.fwd.fit1.whlp15)

lm.fwd.fit1.whlp15.ncv1=lm((GDP.growth)~`64`+`67`+`71`+`74`+`103`+`126`+`165`+`187`+`190`,data=Data)
ncvTest(lm.fwd.fit1.whlp15.ncv1)
plot(lm.fwd.fit1.whlp15.ncv1)

Thus at p=0.011498 null hypothesis cant be rejected hence we will select GDP.growth as response.

##High leverage point:

Data.whlp2 <- Data[-c(15,16,20,6),]

##FORWARD MODEL TEST MSE

set.seed(1)
train=sample(1:nrow(Data.whlp2),size=0.8*nrow(Data.whlp2))
Data.train=Data.whlp2[train,]
Data.test=Data.whlp2[-train,]
dim(Data.train)

dim(Data.test)

lm.fwd.fit1.whlp15.ncv1=lm((GDP.growth)~`64`+`67`+`71`+`74`+`103`+`126`+`165`+`187`+`190`,data=Data.tr
ain)
summary(lm.fwd.fit1.whlp15.ncv1)
pred=predict(lm.fwd.fit1.whlp15.ncv1,Data.test)
mse=mean((pred-Data.test$GDP.growth)^2)
mse

plot(Data.test$GDP.growth,pred,xlab="GDP Growth Observations",ylab="GDP Growth Predictions")
abline(lm.fwd.fit1.whlp15.ncv1,type='l',col='blue')
title(main="Model Accuracy")


## Ridge Regression:

library(glmnet)
library(Matrix)
library(foreach)
x=model.matrix(GDP.growth~.,Data)
y=Data$GDP.growth
grid=10^seq(10,-2,length=100)
```

```
set.seed(1)
train2<-sample(1:nrow(Data),size=0.8*nrow(Data))
test=(-train2)
y.test=y[test]

ridge.tr<-glmnet(x[train2,],y[train2],alpha=0,lambda=grid)
plot(ridge.tr)

cv.out=cv.glmnet(x[train2,],y[train2],alpha=0)
plot(cv.out)
bestlam=cv.out$lambda.min
bestlam

ridge.pred=predict(ridge.tr,s=bestlam,newx=x[test,])
mean((ridge.pred-y.test)^2)

out=glmnet(x,y,alpha=0)
ridge.coef=predict(out, type="coefficients",s=bestlam)[1:190,]
ncoef=as.matrix(ridge.coef)
dim(ncoef)
ncoef
coef.Ridge.Reg=ncoef[-1,]
plot(coef.Ridge.Reg)

x.n=model.matrix(GDP.growth~`3`+`8`+`18`+`28`+`30`+`31`+`32`+`34`+`36`+`39`+`40`+`41`+`42`+`43`+`44`+`
45`+`46`+`47`+`48`+`49`+`51`+`53`+`55`+`56`+`57`+`58`+`59`+`60`+`61`+`62`+`63`+`64`+`65`+`66`+`67`+`68`+
`69`+`70`+`71`+`72`+`73`+`74`+`75`+`77`+`79`+`82`+`83`+`84`+`85`+`86`+`87`+`89`+`91`+`93`+`94`+`130`+`13
2`+`136`+`145`+`153`+`160`+`165`+`171`+`176`+`182`+`187`,Data)

summary(x.n)

y.n=Data$GDP.growth
grid.n=10^seq(10,-2,length=100)

set.seed(1)
train.n<-sample(1:nrow(Data),size=0.8*nrow(Data))
train.n
test.n=(-train.n)
y.test.n=y[test.n]
ridge.tr.n<-glmnet(x.n[train.n,],y.n[train.n],alpha=0,lambda=grid.n)
plot(ridge.tr.n)
summary(ridge.tr.n)

cv.out.n=cv.glmnet(x.n[train.n,],y.n[train.n],alpha=0)
plot(cv.out.n)
bestlam.n=cv.out.n$lambda.min
bestlam.n

ridge.pred.n=predict(ridge.tr.n,s=bestlam.n,newx=x.n[test.n,])
mean((ridge.pred.n-y.test.n)^2)

out.n=glmnet(x,y,alpha=0)
ridge.coef.n=predict(out.n, type="coefficients",s=bestlam.n)[1:190,]
ncoef.n=as.matrix(ridge.coef.n)
dim(ncoef.n)
ncoef.ridge=ncoef.n[-1,]
plot(ncoef.ridge)
```

LASSO regression:

```
library(glmnet)
x=model.matrix(GDP.growth~.,Data)
y=Data$GDP.growth
grid=10^seq(10,-2,length=100)
lasso.mod=glmnet(x,y,alpha=1,lambda=grid)
dim(coef(lasso.mod))
plot(lasso.mod)

set.seed(1)
train2<-sample(seq(58),size=45,replace=FALSE)
test2=(-train2)
y.test=y[test2]
lasso.tr<-glmnet(x[train2,],y[train2],alpha=1,lambda=grid)

set.seed(1)
cv.lasso<-cv.glmnet(x[train2,],y[train2],alpha=1)
plot(cv.lasso)
bestlam=cv.lasso$lambda.min
bestlam
lasso.pred=predict(lasso.tr,s=bestlam,newx=x[test2,])
mean((lasso.pred-y.test)^2)
out=glmnet(x,y,alpha=1,lambda=grid)
lasso.coef=predict(out, type="coefficients",s=bestlam)[1:190,]

lasso.coef[lasso.coef!=0]

lm.fwd.fit1=lm(GDP.growth~`69`+`110`+`127`+`150`,data=Data)
summary(lm.fwd.fit1)

hlp=cooks.distance(lm.fwd.fit1)>0.333
hlp

vif(lm.fwd.fit1)
max(vif(lm.fwd.fit1))

lm.fwd.fit12=lm(GDP.growth~`69`+`127`+`150`,data=Data)
summary(lm.fwd.fit12)

vif(lm.fwd.fit12)
max(vif(lm.fwd.fit12))

set.seed (1)
Data.train.mse=sample(seq(58),size=41,replace=FALSE)
Data.test.mse=Data[-Data.train.mse,]
lm.fit12.mse=predict(lm.fwd.fit12,newdata=Data[-Data.train.mse,])
mean(( lm.fit12.mse-Data.test.mse$GDP.growth)^2)

library(boot)
set.seed(1)
cv.error.10 = rep(0,10)
for (i in 1:10) {
glm.fit = glm(GDP.growth~`69`+`127`+`150`,data=Data)
cv.error.10[i] = cv.glm(data=Data,glm.fit,K=10)$delta[1]
}
```

```
cv.error.10
mean(cv.error.10)
plot(cv.error.10)

## Random Forest:
library (randomForest)
Data1<-read.csv(file.choose(),header = T)

Data1<-Data1[,-1]

missingdata<-as.data.frame(sort(sapply(Data1, function(x) sum(is.na(x))),decreasing=T))
missingdata<-(missingdata/nrow(Data1))*100
class(missingdata)
colnames(missingdata)[1]<-"Percentage"
missingdata$predictors<-row.names(missingdata)

x<-subset(missingdata,missingdata$Percentage==0)

Data1<-Data1[,(x$predictors)]

set.seed(1)
train<-sample(1:nrow(Data1),size=0.8*nrow(Data1))
Data.train<-Data1[train,]
Data.test<-Data1[-train,]
dim(Data.train)
dim(Data.test)
sum(is.na(Data1))

rf<-randomForest(GDP.growth~.,data = Data.train,mtry=63,importance=TRUE)

yhat<-predict(rf,newdata=Data.test)
mean(( yhat-Data.test$GDP.growth)^2)
plot(rf)
importance(rf)
varImpPlot(rf)

##PCA:
PCA<-prcomp(Data,center = T,scale. = T)

library(factoextra)
fviz_pca_var(PCA, col.var = "blue")
fviz_eig(PCA)

library(pls)
Y<-Data1$GDP.growth
PCR<-pcr(GDP.growth~.,data=Data.train,scale = TRUE, validation = "CV")
pcr_pred <- predict(PCR, Data.test, ncomp = 2)
mean((pcr_pred - Data.test$GDP.growth)^2)

summary(PCR)
colnames(Data1)
```