FORECASTING MODELS: PCE CONSUMPTION

[Document subtitle]

ABSTRACT

This study analyses **U.S. Personal Consumption Expenditure (PCE) data** using three forecasting models: **Drift Method, Holt's Exponential Smoothing, and Auto ARIMA**. After **data cleaning, missing value imputation, and trend analysis**, each model was trained and tested for accuracy using **RMSE and MAE metrics**. Results indicate that **Holt's Exponential Smoothing** outperformed other models, providing the most reliable forecasts. The study highlights the **effectiveness of time series forecasting techniques** in predicting future consumption trends, aiding economic planning and decision-making.

Shreshth Sharma

Table of Contents

Note: *Check For explanation under Appendices Section for yellow highlighted data.*

US Personal Consumption Data – Forecasting Models

1. Introduction

In this analysis, we use the seasonally-adjusted PCE data from the United States, sourced from a CSV file named "PCE.csv," to compare the predictive capabilities of three distinct forecasting models.

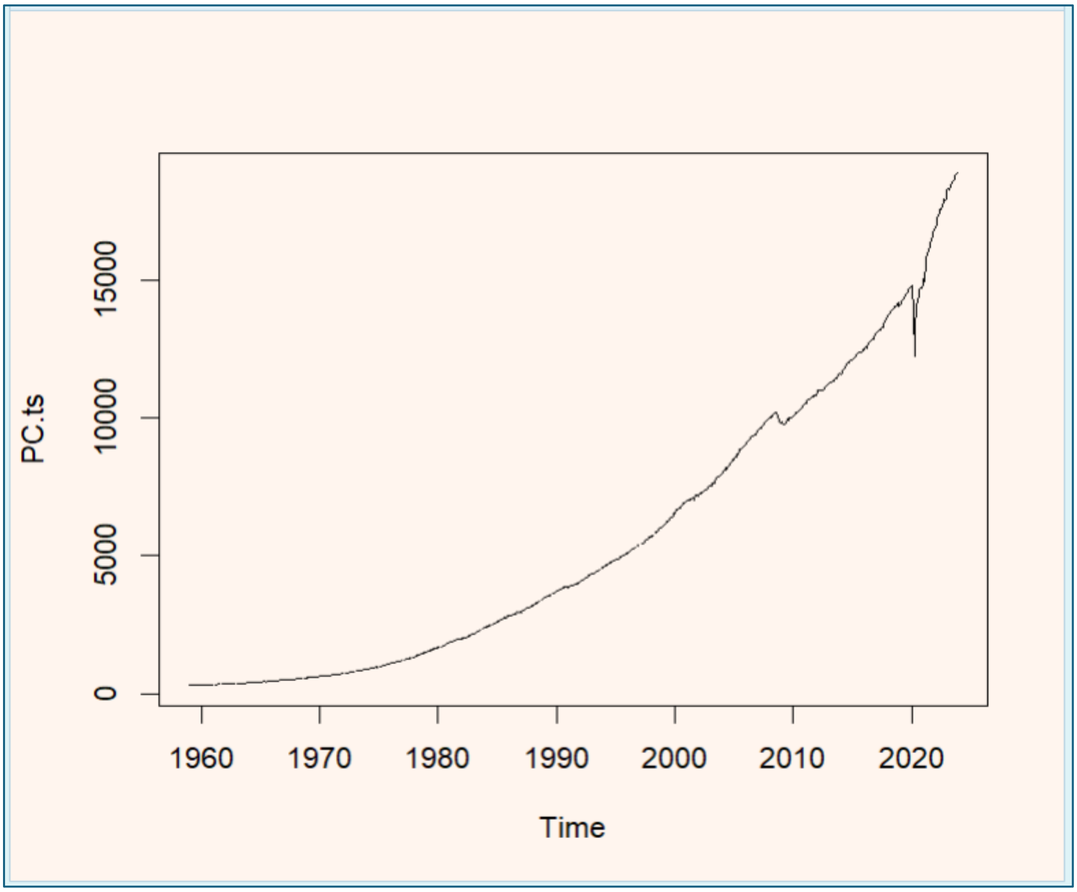
2. Data Preparation:
The analysis begins with loading the PCE data from the "PCE.csv" file. Initial steps include:

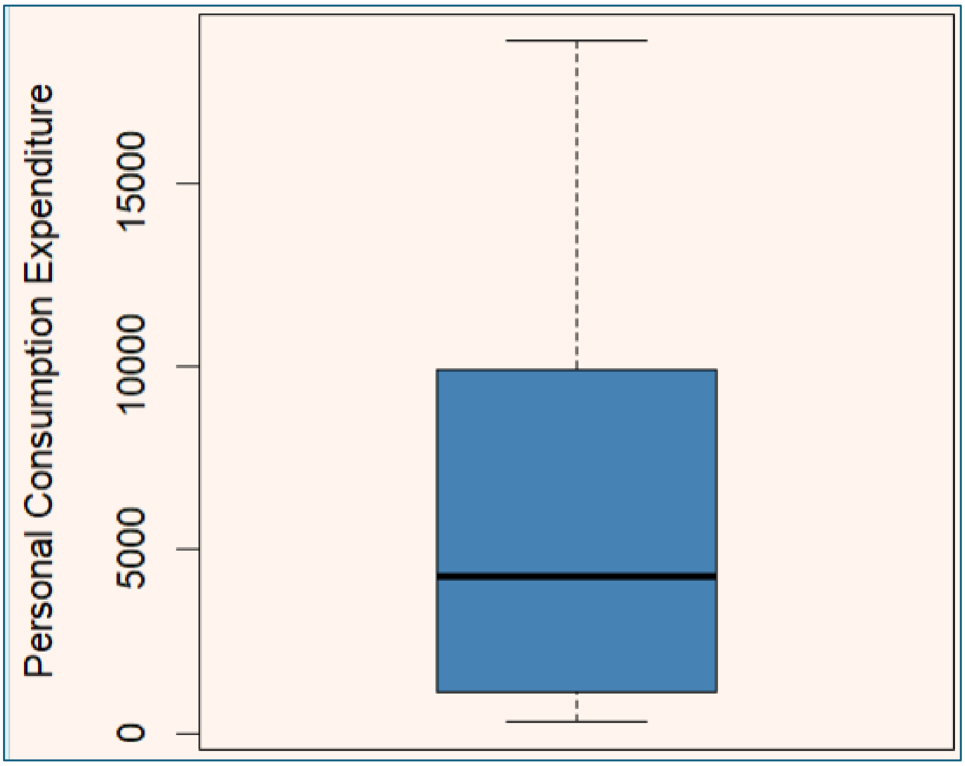
2.1 Data Cleaning and Data Visualization:
Checking for and handling missing values, outliers, or erroneous entries. Plotting the data to understand its characteristics, such as trends and seasonality.

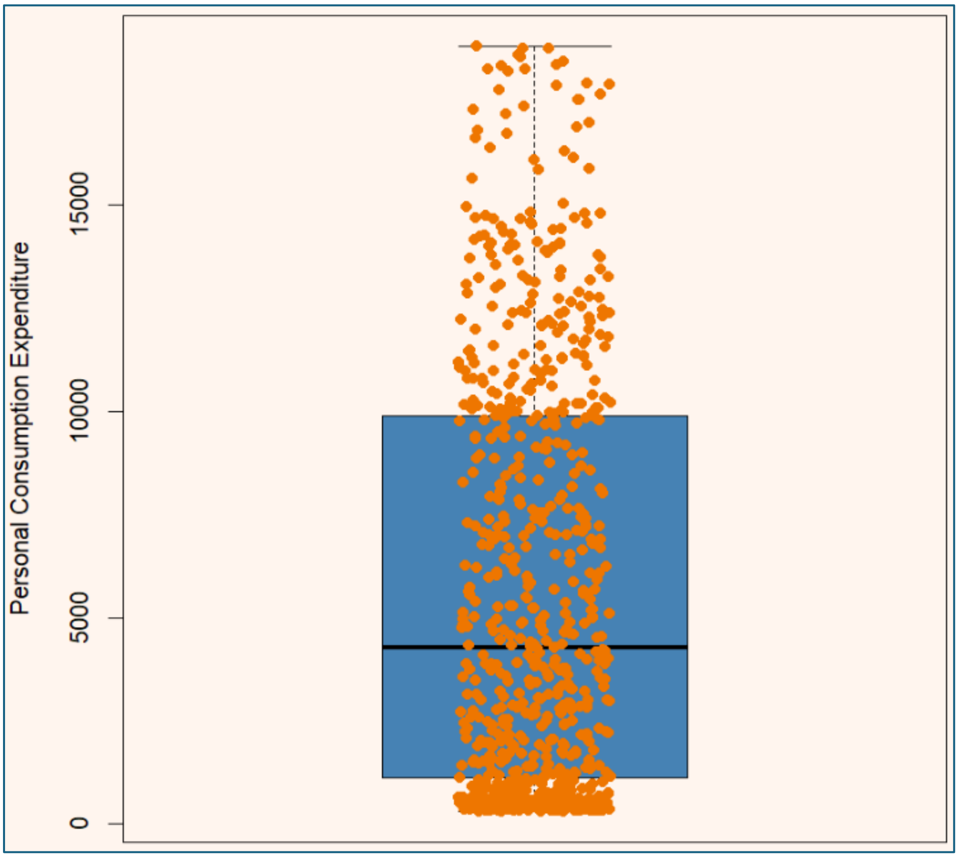
Summary for Dataset

```
> summary(PCdata)
     DATE                PCE        
 Length:779         Min.   :  306.1  
 Class :character   1st Qu.: 1124.7  
 Mode  :character   Median : 4270.0  
                    Mean   : 5792.1  
                    3rd Qu.: 9896.7  
                    Max.   :18858.9  
                    NA's   :43       
```

Summary of Personal Consumption Expenditure states the data has **779 observations** with 2 columns as “DATE” (Month of Observation) and “PCE” (Personal Consumption Expenditure).







Box Plot – PCE for US consumers.

In the Box-Plot above we can see the median value is close to 5000 and it is evident that more than 50% of data lies below 5000.

In column PCE the minimum and maximum recorded value is **306.1** and **18858.9** with a mean value at **5792.1**. This raise suspicion about the major data being near to initial values and in later years the consumption Expenditure have experienced a dramatic increase.

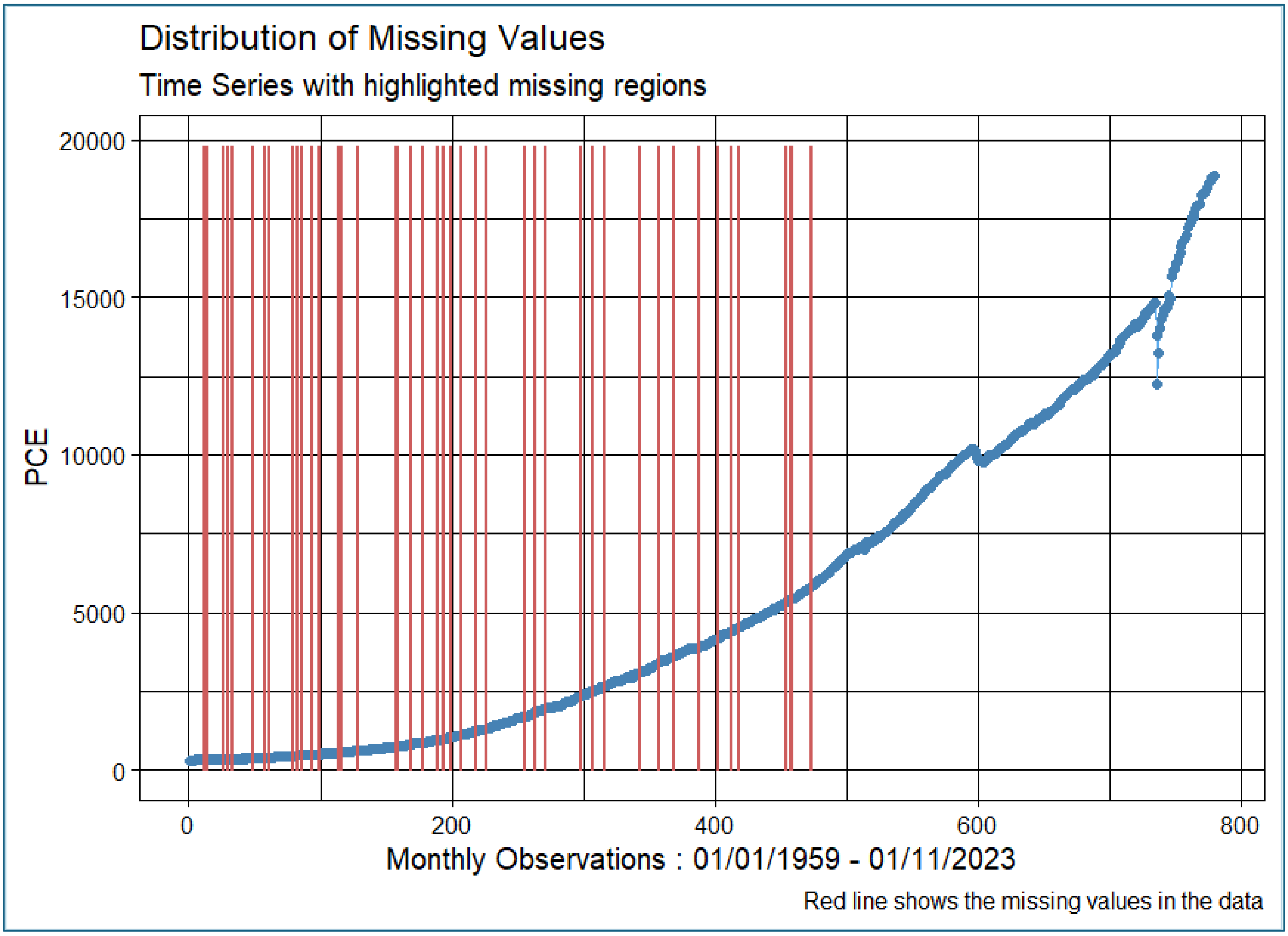
Upon Critical Observation we can observe there are no outliers. And majorly the data is spread in the lower section of Box-Plot.

```
> outlier_indices
integer(0)
```

Missing Data:

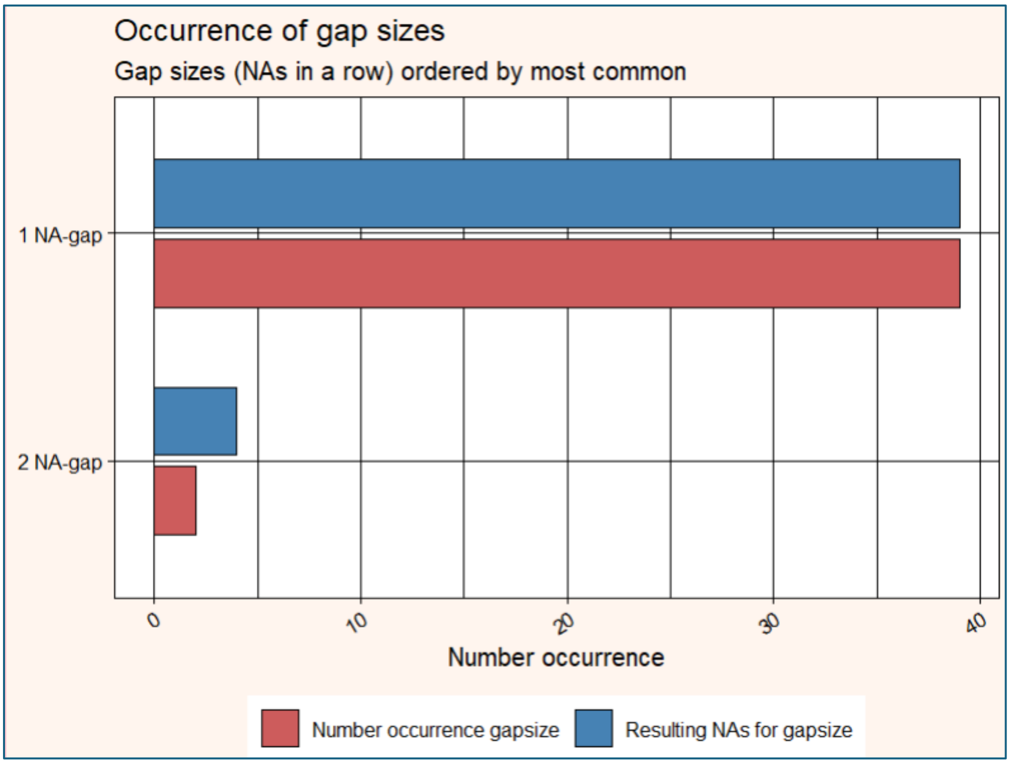
From the summary, section we can see that there are **43 *NA's*** values in the PCE column.

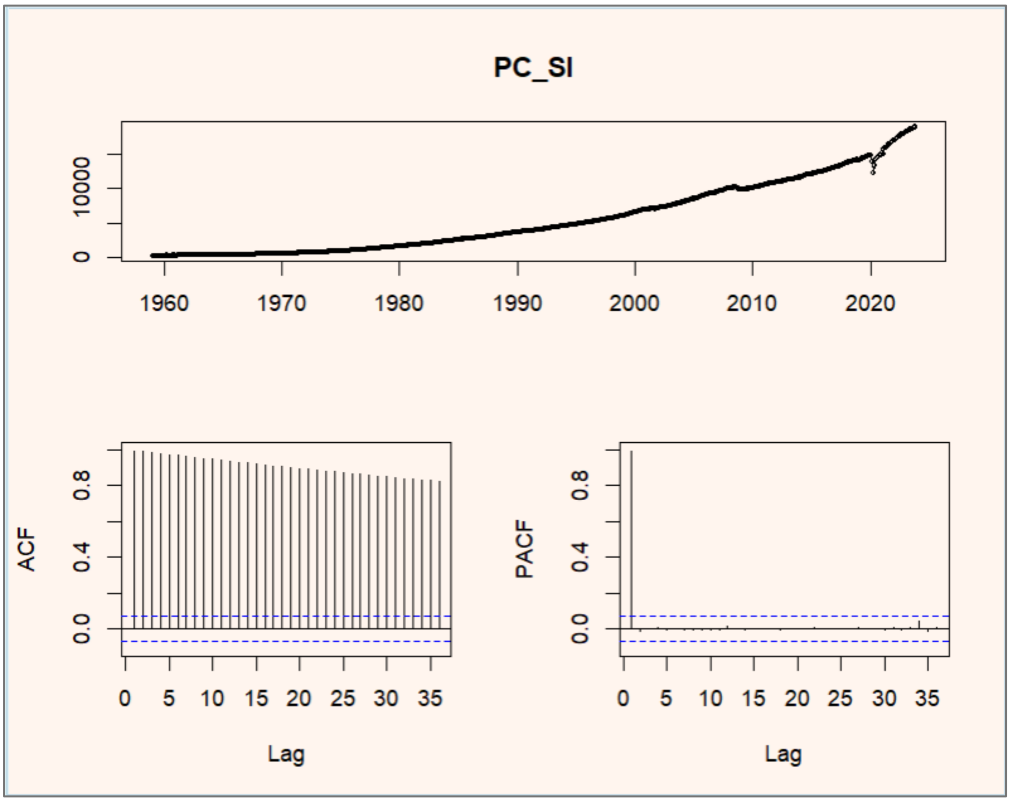
```
> sum(complete.cases(PCdata))
[1] 736
> #Missing Values
> sum(!complete.cases(PCdata))
[1] 43
```



There are many ways to treat missing data, like deletion and imputation. Deletion would account to **loss of 5.5%** of data which would influence our analysis negatively.

Imputing missing values in time series data is a crucial step in data preprocessing to ensure accurate analysis and forecasting. In R Impute-Package specializes in univariate time series imputation, providing various techniques such as linear/nonlinear interpolation, decompositions, and Kalman filtering to fill irregularly spaced series gaps (Nickolas & Shobha, 2021).





Pre-Processing

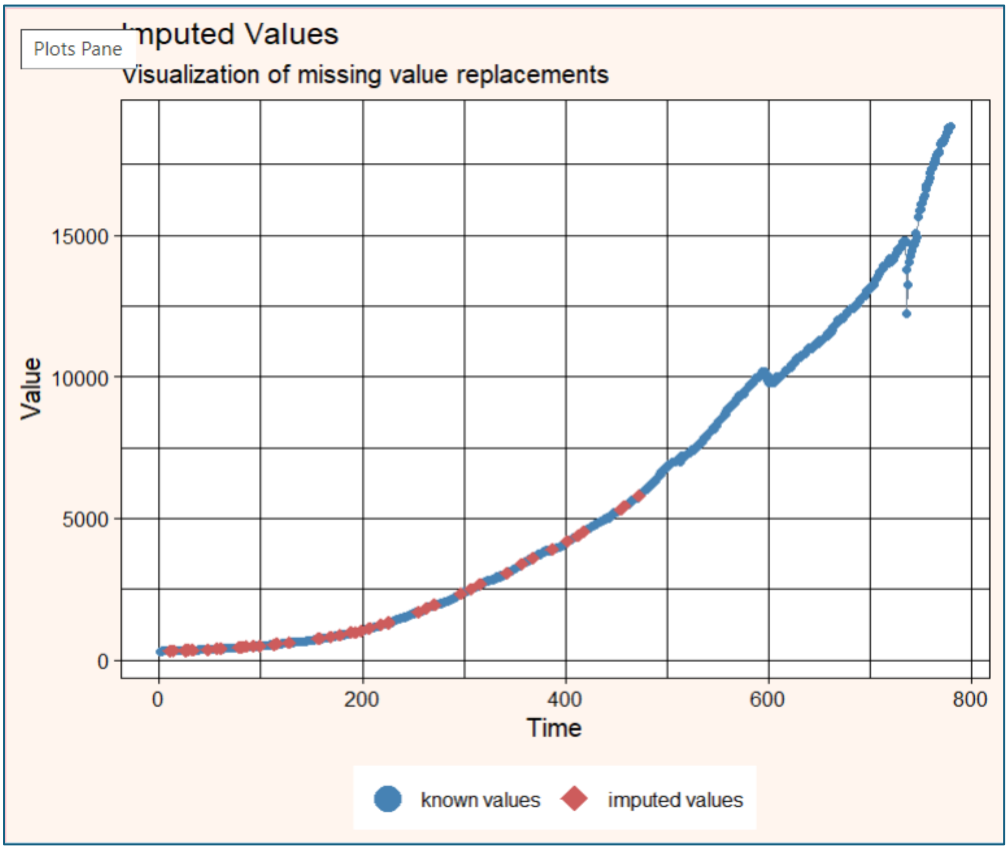
Analysing from the gap size plot we can conclude that there are majorly 1-gap i.e. 39 NAs at a time between data and with just two 2-gap missing data resulting in a total gap of 4 NAs.

Also, It can be observed that under ACF plots values are way above the significance boundaries. Hence, can be predicted well enough even after 35-lags. This shows there's a significant correlation between values.

Handling Missing Data: For linear trends we can use simple interpolation or kalman. na_kalman() is used when you need a more sophisticated approach that can capture complex patterns particularly for high-frequency time series data.

Abb.	Missing Data Treatment
PC.ts	Time Series
PC_SI	Simple Interpolation
PC_MA	Moving Average
PC_MAW	Weighted Moving Average
PC_KL	Kalman
PC_KA	Kalman Auto Arima

S. No.	PC.ts	PC_SI	PC_MA	PC_MAW	PC_KL	PC_KA
48	NA	373.05	371.1232	371.7493	370.8385	373.7085
49	374.4	374.4	374.4	374.4	374.4	374.4
50	373.4	373.4	373.4	373.4	373.4	373.4
51	375	375	375	375	375	375
52	376.4	376.4	376.4	376.4	376.4	376.4
53	377.2	377.2	377.2	377.2	377.2	377.2
54	381.7	381.7	381.7	381.7	381.7	381.7
55	384.4	384.4	384.4	384.4	384.4	384.4
56	386.3	386.3	386.3	386.3	386.3	386.3
57	NA	386.15	388.2614	386.4996	386.1947	385.3762
58	386	386	386	386	386	386



Handling Missing Data

Method Chosen: Simple Interpolation

Reason:

1. As the trend is quite linear and there seems to be no seasonality.

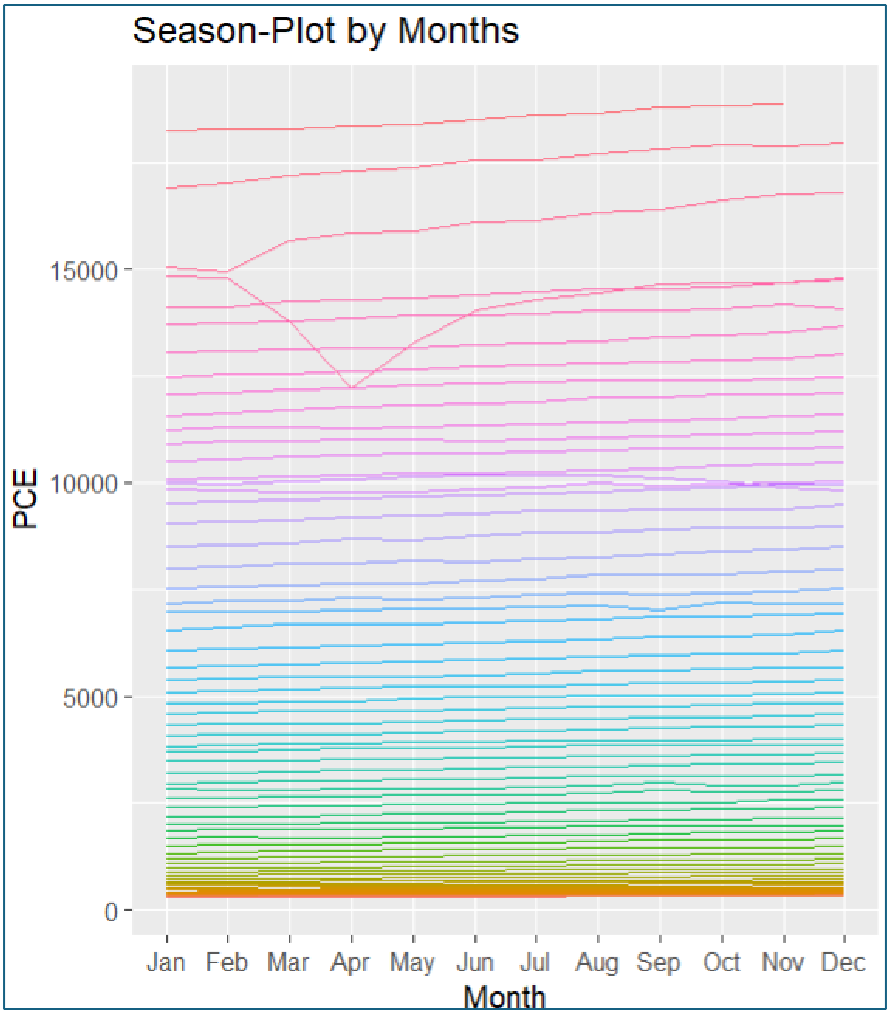
2. Missing values are infrequent and not systematic.

3. Upon comparison Simple-Interpolation offers best imputations for missing values.

4. Save Computational-Overheads comparing to other methods.

Imputed Values stored in the dataset fits well in the graph.

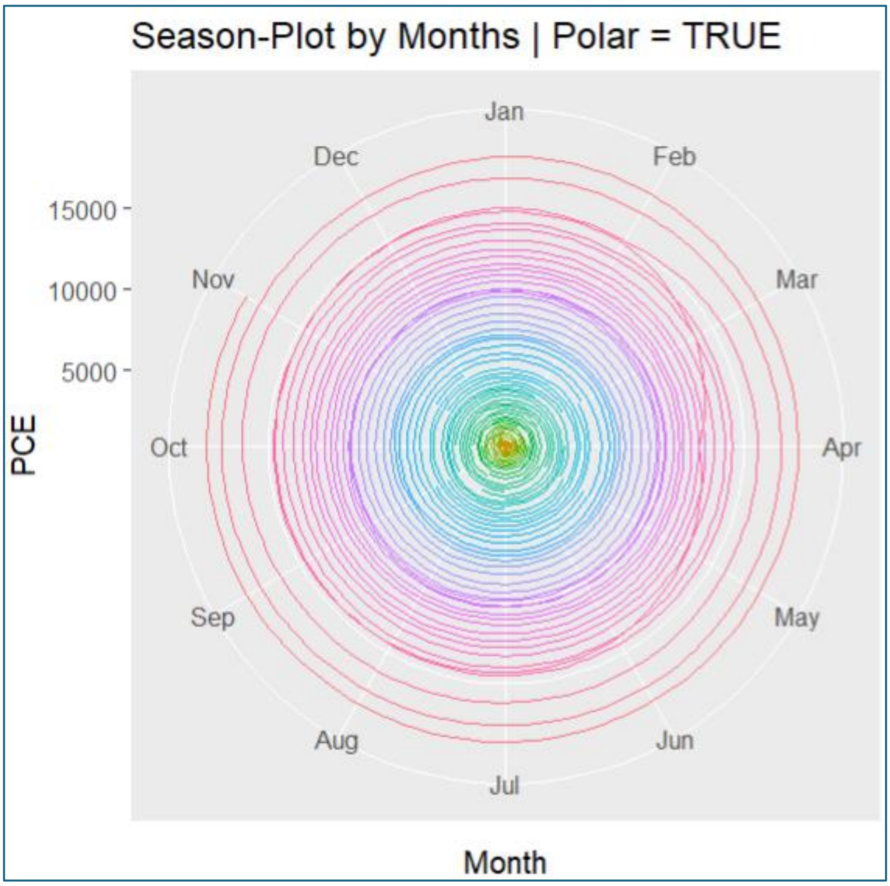
Season Plot.

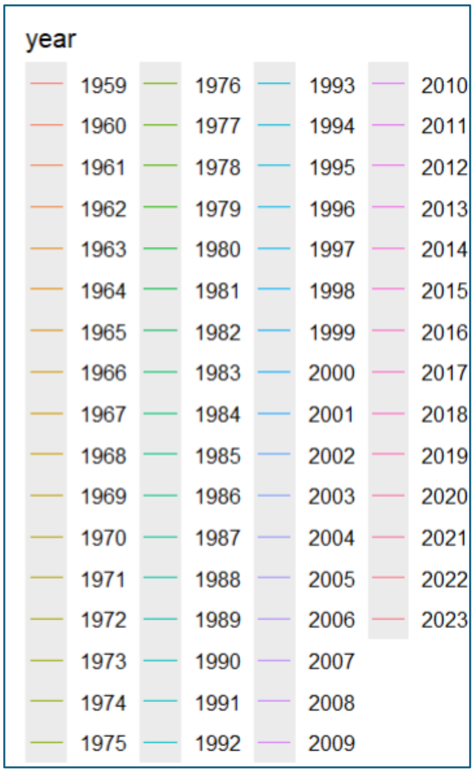


Seasonality

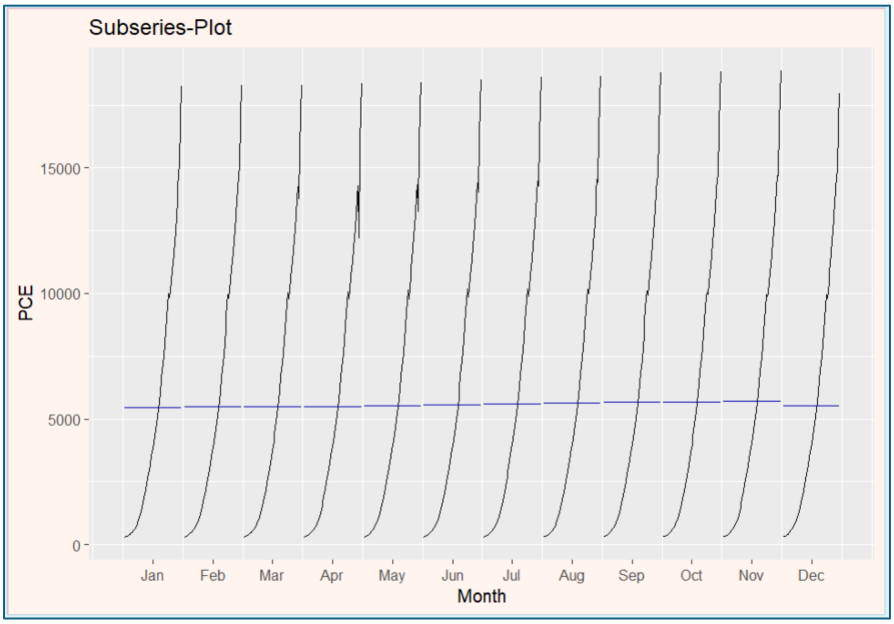
PCE has exhibited quite a linear trend with respect to seasons and months.

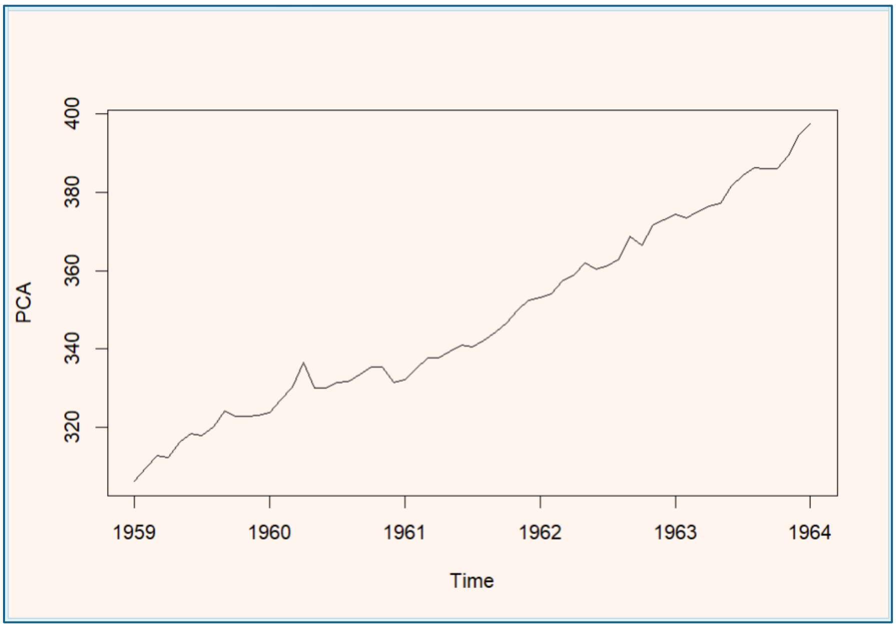
So, we can declare the data has **no seasonality** over the given time-period.

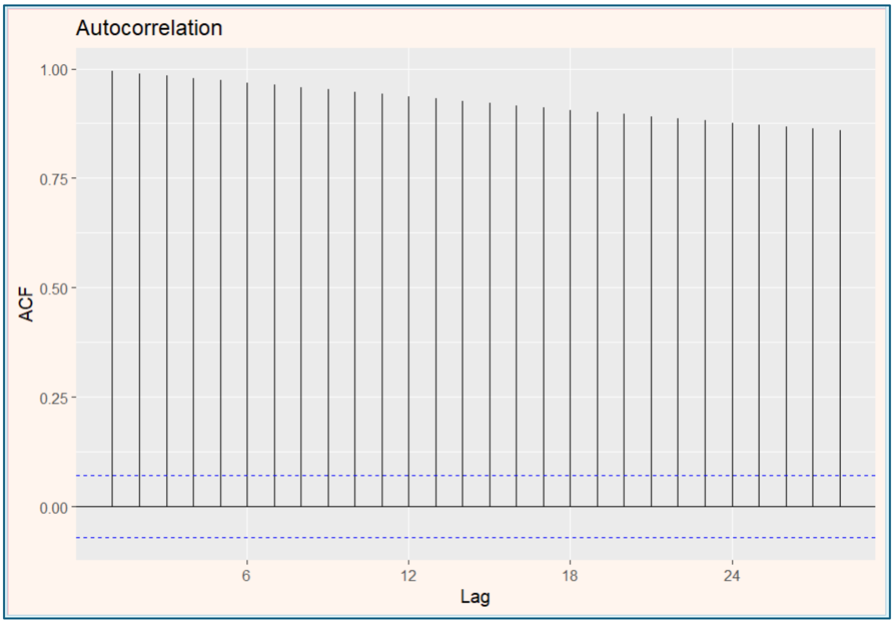




Sub-Series & ACF Plots:







Cyclicity

For PCE over the years there is a linear trend across all months with no major fluctuation showing no sign of cyclicity in the data.

With Autocorrelation-Plot we can observe that every data is closely co-related to next coming data and there is a linear trend of correlation between the data points.

2.2 Method Selection & Training:

Here PC_SI is the personal consumption expenditure time series with missing values imputed using function na_interpolation().

Method Selection Process:

- **Simple Forecasting:** The drift method is applied to time-series data where the trend is continuous, and it helps in capturing the underlying linear pattern in the data (Zulkifle et al., 2022).
- **Exponential Smoothening:** Holt's method is effective for forecasting trends in time series data and is widely used in practical fields for its ability to handle trended time series (Chatfield, 1978).
- **ARIMA models:** Auto ARIMA has been shown to outperform manual ARIMA in terms of determining the appropriate ARIMA parameters(p, d, q), based on measures such as root mean square error (RMSE), mean absolute error (MAE) without a manual intervention of an expert data scientist (Al-Qazzaz & Yousif, 2022).

Train Set represents data trained for the models using **80%** of the initial data of the imputed time series.

Train set is represented by **"train".** Inside variable **"train"** we have stored a subset of our imputed time series **"PC_SI". Here "end = 620"** denoting the subset has first 620 observations out of 779 observations.

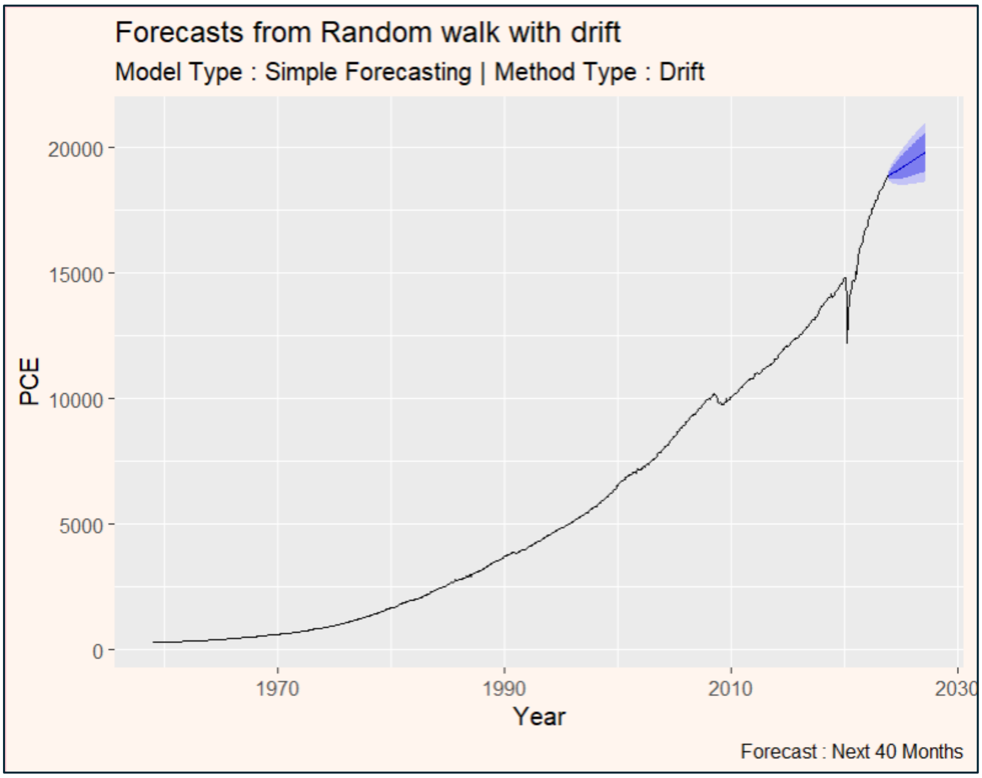
Test Set:

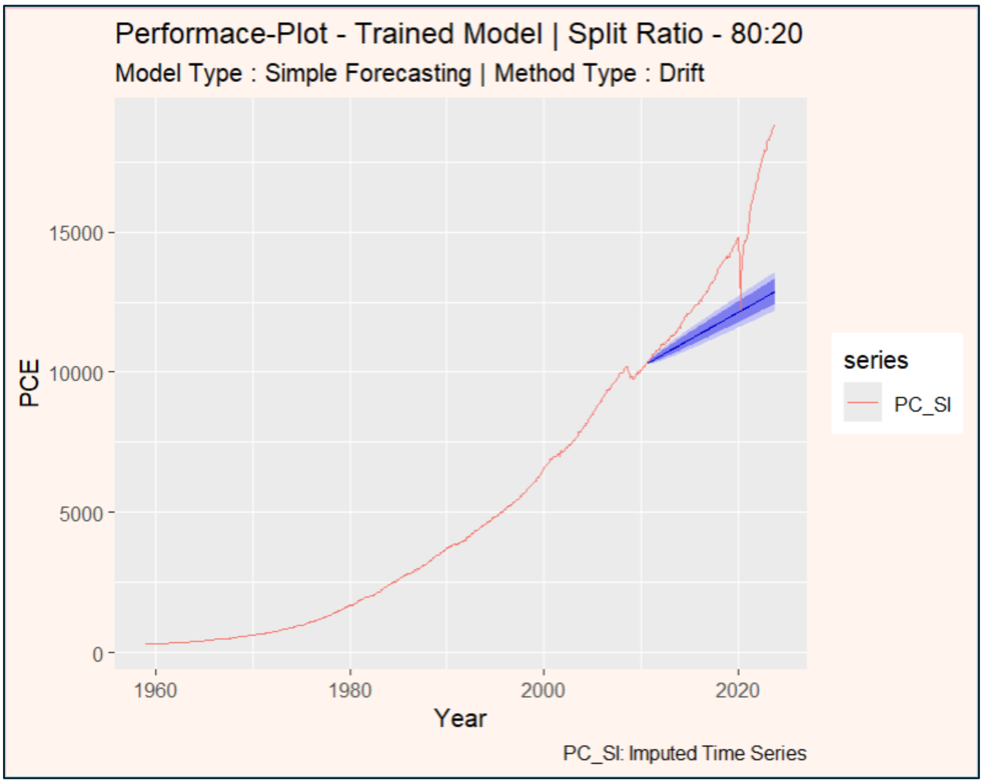
Using next **20% observations** would be stored in the test set over which model accuracy will be tested. The observation under test sets would be compared with the imputed time series to find the best performing models out of Drift, Holt's and auto.arima. **"test_d"** denotes the test data stored using drift method whereas **"test_h"** denotes holt's and **"test_a"** represents data stored using auto.arima method.

```
# - Train Set - #
train <- subset(PC_SI, end = 620)
# - Train and Test Drift Method - #
test_d <- rwf(train, h = 159, drift = TRUE)
# -  Train and Test Holt's Method - #
test_h <- holt(train, h = 159)
# -  Train and Test Arima Method - #
# - Train Arima - #
train_a <- auto.arima(train)
# - Testing Model - #
test_a <- forecast(train_a, h = 159)
```

3. Model Development and Selection

3.1 Simple Forecasting: Drift method





Simple Forecasting: Drift's Method

Reason:

As our data-trend is quite linear and The drift method of forecasting is beneficial for capturing linear trends in data *Pwasong & Sathasivam (2015).*

Observation:

In the first plot we can see the forecast done for next 40 periods are quite acceptable and somehow tries to justify the trend.

Testing:

While testing drift after training model it **doesn't capture trend properly.**

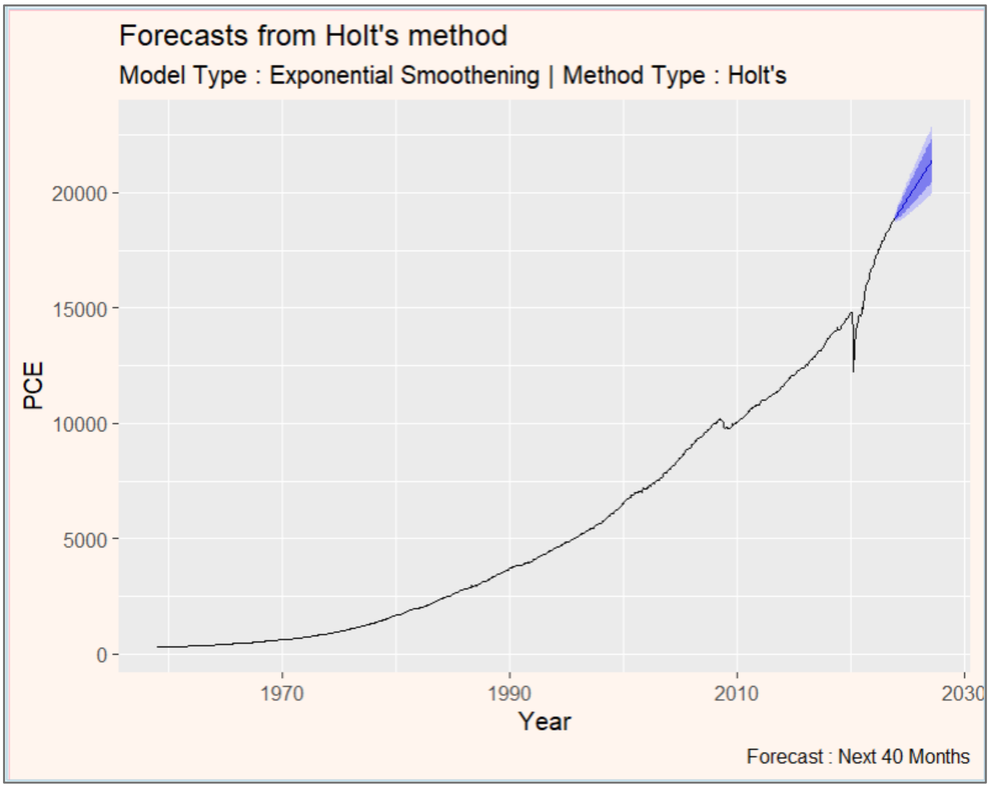
Accuracy:

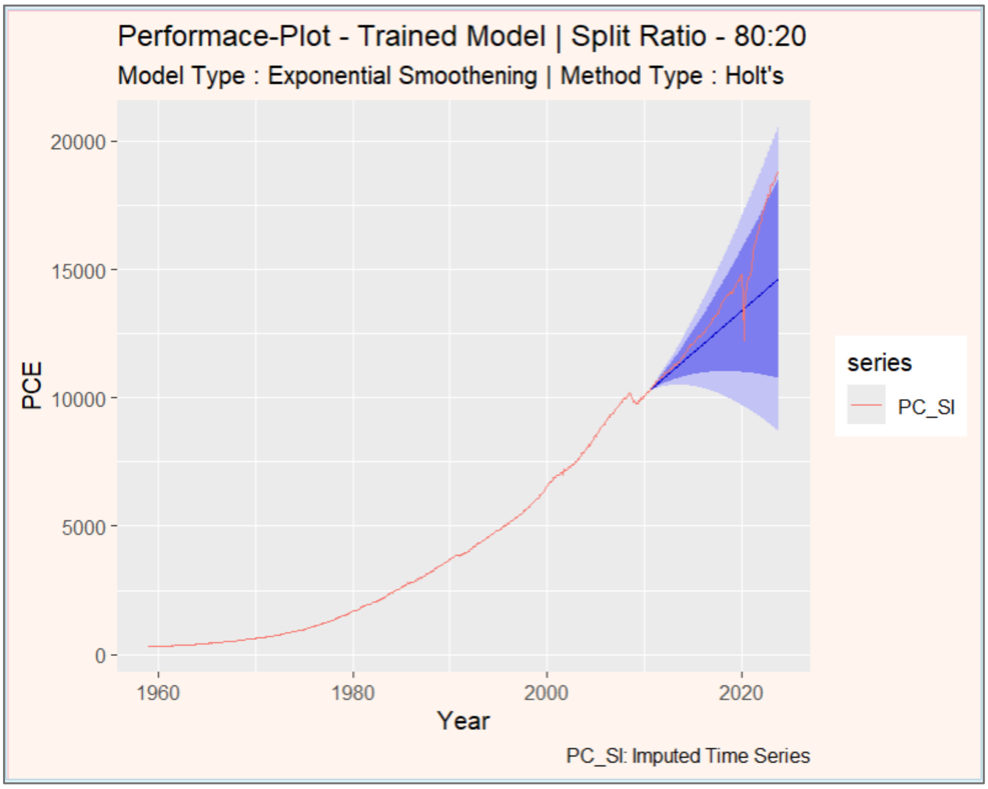
Model accuracy can be measured by **RMSE and MAE values** in the green cubical shapes.

```
> accuracy(test_d, PC_SI)
                        ME       RMSE       MAE        MPE      MAPE       MASE      ACF1
Training set -8.440539e-14   24.80443   16.6533 -0.8227941  1.127426 0.08255875 0.1173835
Test set      1.976687e+03 2590.92437 1976.6869 13.0008764 13.000876 9.79942629 0.9704643
             Theil's U
Training set        NA
Test set      10.71193
```

3.2 Exponential Smoothening Method: Holt Method

3.2 Exponential Smoothening Method: Holt Method: Holt's method involves two smoothing parameters, α and γ, which allow for capturing both the level and trend in the data *Trull et al. (2020).*





Exponential Smoothening: Holt's Method

Reason:

Holt's Method is commonly used for forecasting trends in time series data that exhibit a trend component. *(Oni & Akanle, 2018).*

Observation:

In the first plot we can see the forecast done for next 40 periods are better than drift and captures trend better.

Testing:

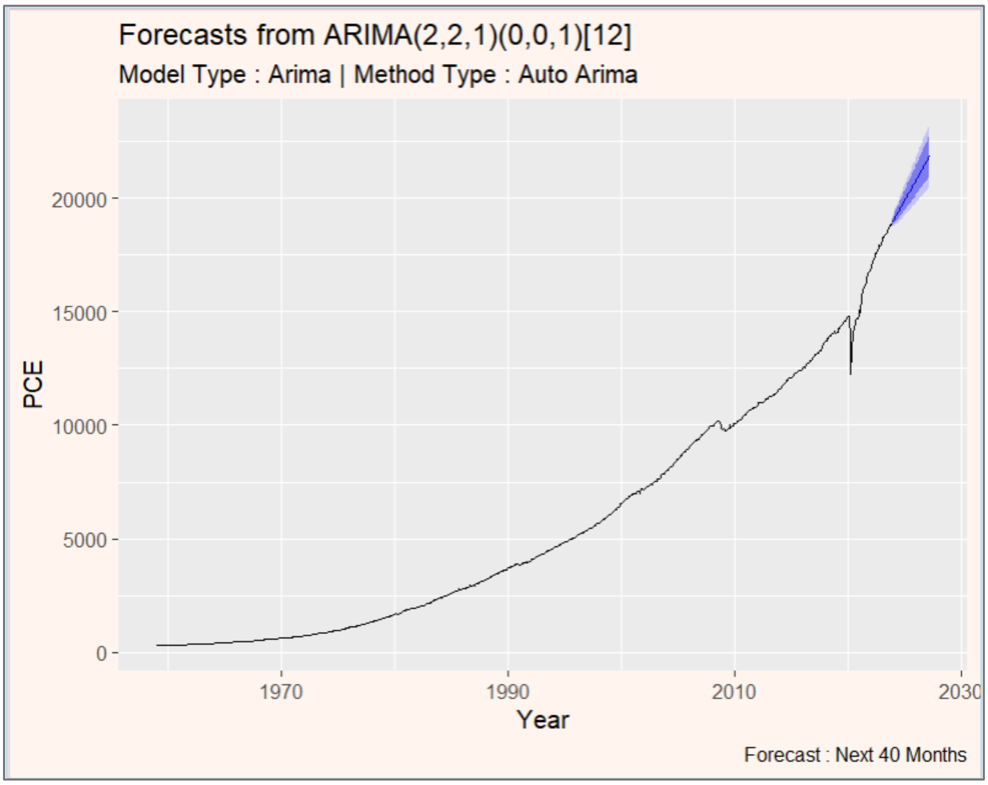
While testing we can observe actual trend lies in the trained model range with **95% confidence interval.**

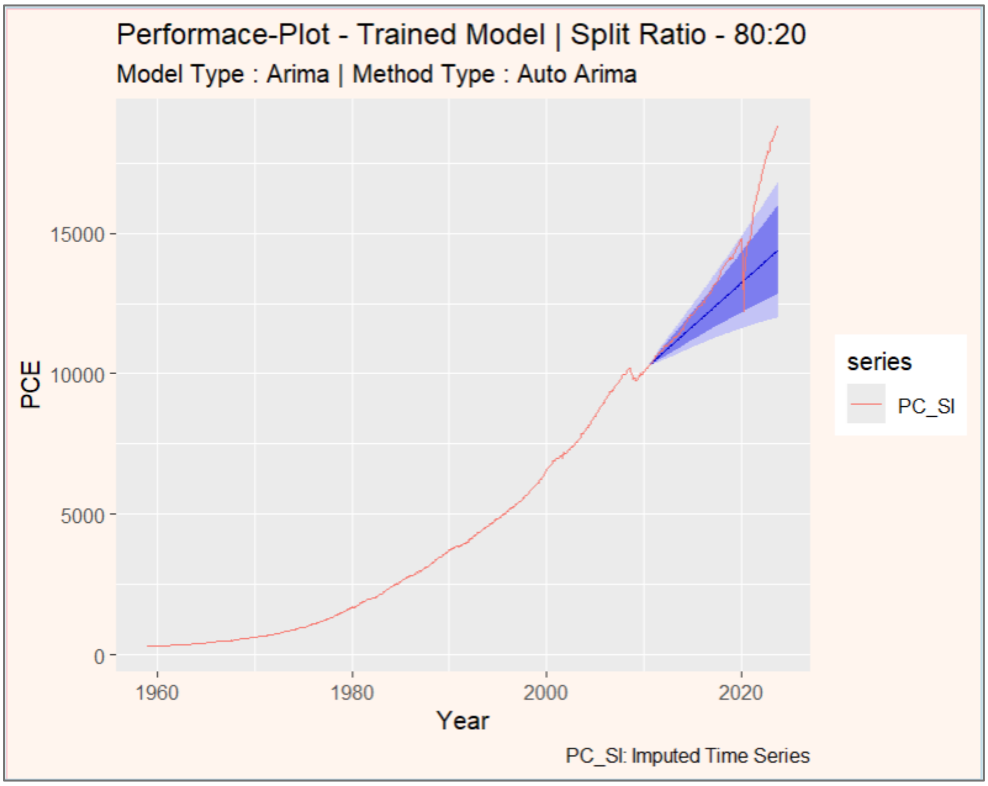
Accuracy:

RMSE and MAE values in the yellow cubical shapes below seems to be better than "drift's".

```
> accuracy(test_h, PC_SI)
                     ME       RMSE        MAE        MPE      MAPE       MASE        ACF1
Training set  0.3821172   22.47675   12.31928 0.02627336 0.3981727 0.06107282 -0.01489117
Test set   1085.5954619 1634.53171 1104.48165 6.86970992 7.0222676 5.47546828  0.96368017
             Theil's U
Training set        NA
Test set      6.498915
```

3.3 Arima Models: Auto Arima – [Auto Regressive order = 2, Differencing Order = 2, Moving Average = 1]





ARIMA: Auto.arima Method

Reason:

Auto Arima performs better than normal ARIMA models.

Observation:

In the first plot we can see the forecast done for next 40 periods are fairly capturing the trend similar to Holt's model. To make series stationary it has been differenced 2 times.

Testing:

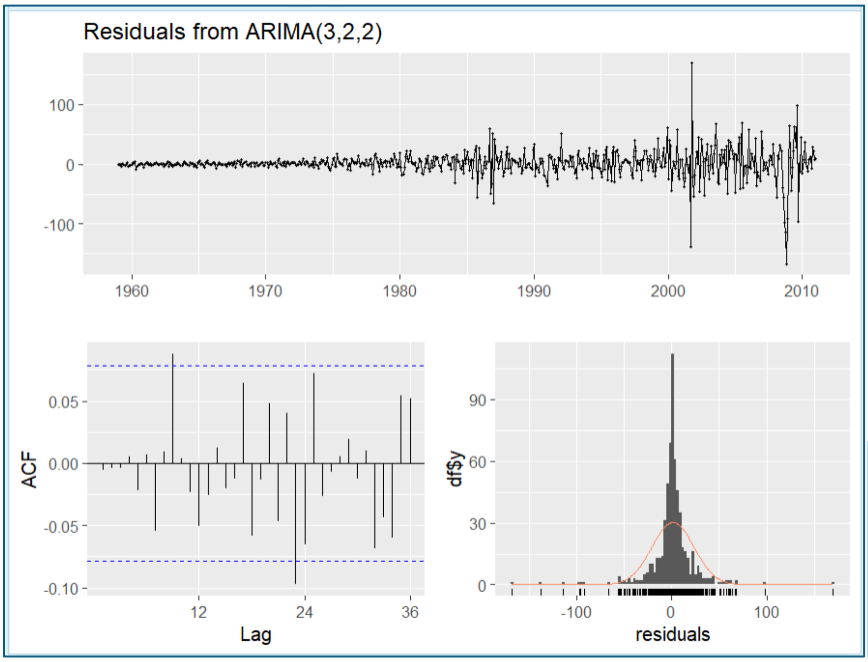
After Training we can observe the auto Arima model seems to capture less trend with the same confidence interval than the Holt's

Accuracy:

RMSE and MAPE in the purple cubical shapes below seems to be more than ones in Holt's.

```
> accuracy(test_a, PC_SI)
                       ME       RMSE        MAE        MPE      MAPE       MASE         ACF1
Training set     1.152621   22.10735   12.32169 0.06569057 0.4037185 0.06108479 -0.005381621
Test set      1193.937042 1752.77032 1208.66832 7.60326703 7.7231955 5.99197371  0.965040211
             Theil's U
Training set        NA
Test set      7.008244
```

Checking Residuals for test set after using ARIMA.




```
> # - Checking Residuals - #
> checkresiduals(test_a)

        Ljung-Box test

data:  Residuals from ARIMA(3,2,2)
Q* = 28.008, df = 19, p-value = 0.08328

Model df: 5.   Total lags used: 24
```

Arima there are just 2 spikes out of the significance level in ACF plot and data is also normally distributed in PACF. It almost represents a white noise pattern and **P-value is significant** so the null hypothesis will be rejected i.e. there is **no correlation between residuals**. Hence, ARIMA models fits good with the data and evidently will perform well while forecasting future values.

4. Model Evaluation: Compare Models

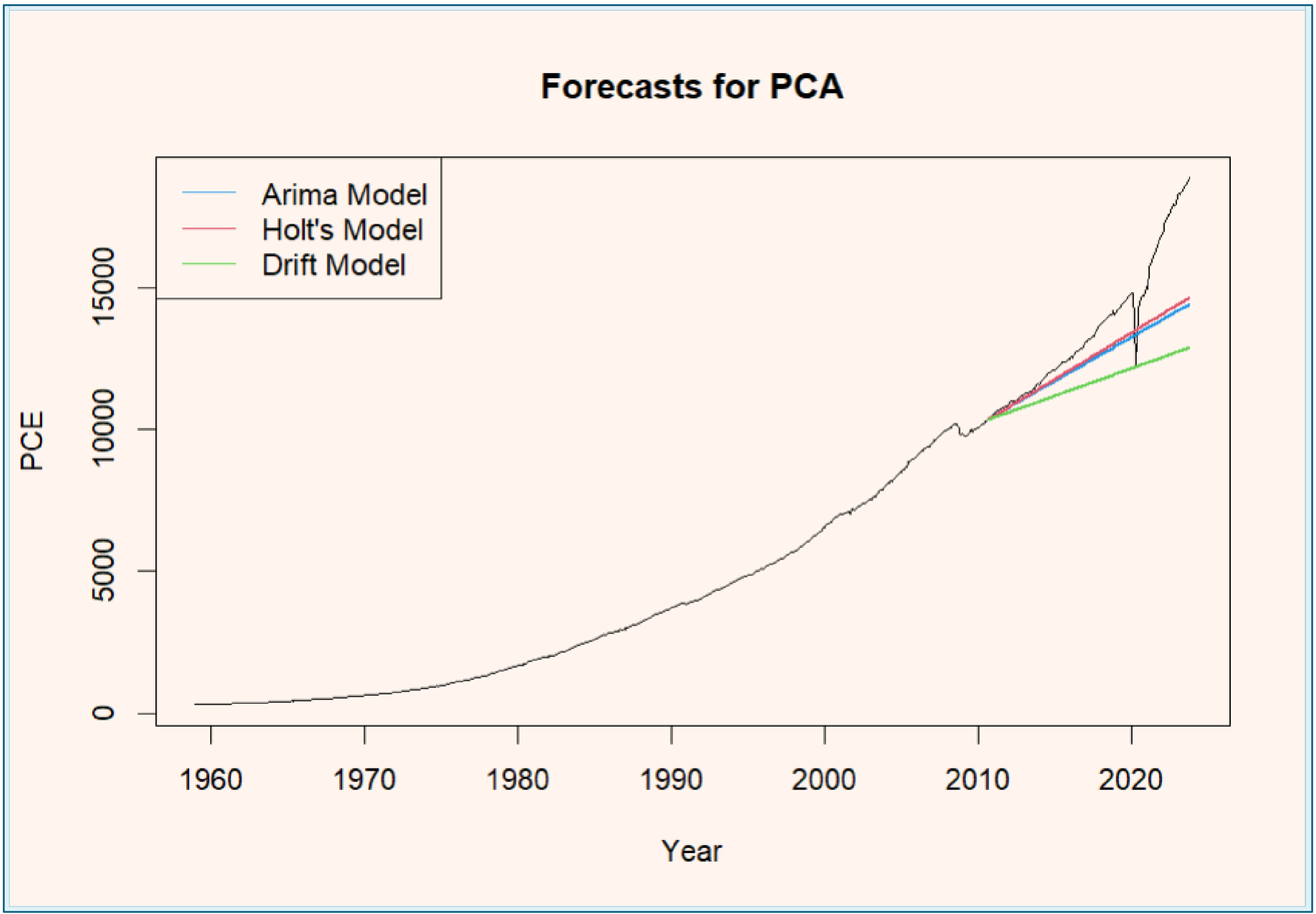
4.1 Forecast Accuracy

Selection Criteria: RMSE (Root of the mean squared errors) and MAE (mean of absolute errors) scores can help in identifying the best model. *Kouadri et al. (2021)* emphasized the significance of RMSE as a predictive numerical index for measuring model performance in time series forecasting. Hence, we are looking for **the models with least values of MAE and RMSE**.

80:20 Split Ratio | Trained Model		
Model	**RMSE**	**MAE**
Arima	1544.7639	994.1764
Holt's	1003.6052	566.5836
Drift	2590.9244	1976.6869

Holt's Model stand out and performs best with the **least RMSE and MAE** values with the test set.

4.2 Model Validation: Graphical analysis to compare the forecasts from each model against the actual data.



4.3 One Step Ahead Rolling Forecast:

```
> #### ------------------------------------------------------------- ####
> # - [ One Step Ahead Rolling Forecast Without Re-estimation ] - #
> library(fpp)
>
> # - [ Drift Model ] - #
> fit_roll_d <- rwf(train_roll)
> refit_roll_d <- rwf(PC_SI, model=fit_roll_d)
> rfd <- window(fitted(refit_roll_d), start=1960)
> accuracy(rfd, PC_SI)
               ME     RMSE      MAE       MPE      MAPE      ACF1 Theil's U
Test set 24.16688 96.79771 34.83651 0.5254067 0.6578525 0.2002451         1
>
> # - [ Holt's Model ] - #
> fit_roll_h <- holt(train_roll)
> refit_roll_h <- holt(PC_SI, model=fit_roll_h)
> rfh <- window(fitted(refit_roll_h), start=1960)
> accuracy(rfh, PC_SI)
               ME     RMSE      MAE       MPE      MAPE      ACF1 Theil's U
Test set 5.842065 92.58682 24.50058 0.1440243 0.4353127 0.1654903 0.8456164
>
> # - [ Arima Model ] - #
> train_roll <- window(PC_SI,end=1959.99)
> fit_roll <- auto.arima(train_roll)
> refit_roll <- Arima(PC_SI, model=fit_roll)
> rfa <- window(fitted(refit_roll), start=1960)
> accuracy(rfa, PC_SI)
               ME     RMSE      MAE       MPE      MAPE      ACF1 Theil's U
Test set 22.63961 96.42774 33.71286 0.4240249 0.5929306 0.2002451 0.9488181
>
```

Results:

One Step Ahead Rolling Forecast		
Model	**RMSE**	**MAE**
Arima	96.4277	33.7128
Holt's	92.5868	24.5005
Drift	96.7977	34.8365

Holt's Model stand out and performs best with the **least RMSE and MAE**.

4.4 Forecast For OCT 2024:

In both cases, Holt's seems to perform better and forecast for

Forecast For October using all models						
Model	**Month**	**Forecast [PCE]**	**Lo 80**	**Hi 80**	**Lo 95**	**Hi 95**
Drift	**Oct-24**	19121.21	18722.43	19520	18511.33	19731.1
Holt's	**Oct-24**	19566.92	19147.56	19986.28	18925.56	20208.28
Arima	**Oct-24**	19682.71	19292.37	20073.04	19085.73	20279.68