TIME SERIES ANALYSIS OF DAILY CRUDE OIL PRODUCTION - A DATA QUALITY ASSESSMENT

Submitted by:
Nidhi Mankala

Open source softwares used:

- Python 3.9
- Jupyter notebook
- Microsoft SQL Server Management Studio 17
- Spotfire 11.5.0
- R studio

Dataset

The dataset contains DCP values for everyday from 1/1/2016 to 16/10/2021

It contains 4 columns

1) LevelType: Field or GC
2) LevelName: RA,SA,AD,RQ, undefined
3) Date
4) DCP Integer

Dataset captured from Microsoft SQL Server

90 %

Results | Messages

	LevelType	LevelName	Date	DCP
1	Field	RA	2021-10-16	151644
2	Field	SA	2021-10-16	85424
3	Field	Undefined	2021-10-16	8355
4	Field	AD	2021-10-15	6783
5	Field	RA	2021-10-15	136145
6	Field	RQ	2021-10-15	9188
7	Field	SA	2021-10-15	93898
8	Field	AD	2021-10-14	6775
9	Field	RA	2021-10-14	135826
10	Field	RQ	2021-10-14	9108

What is time series analysis

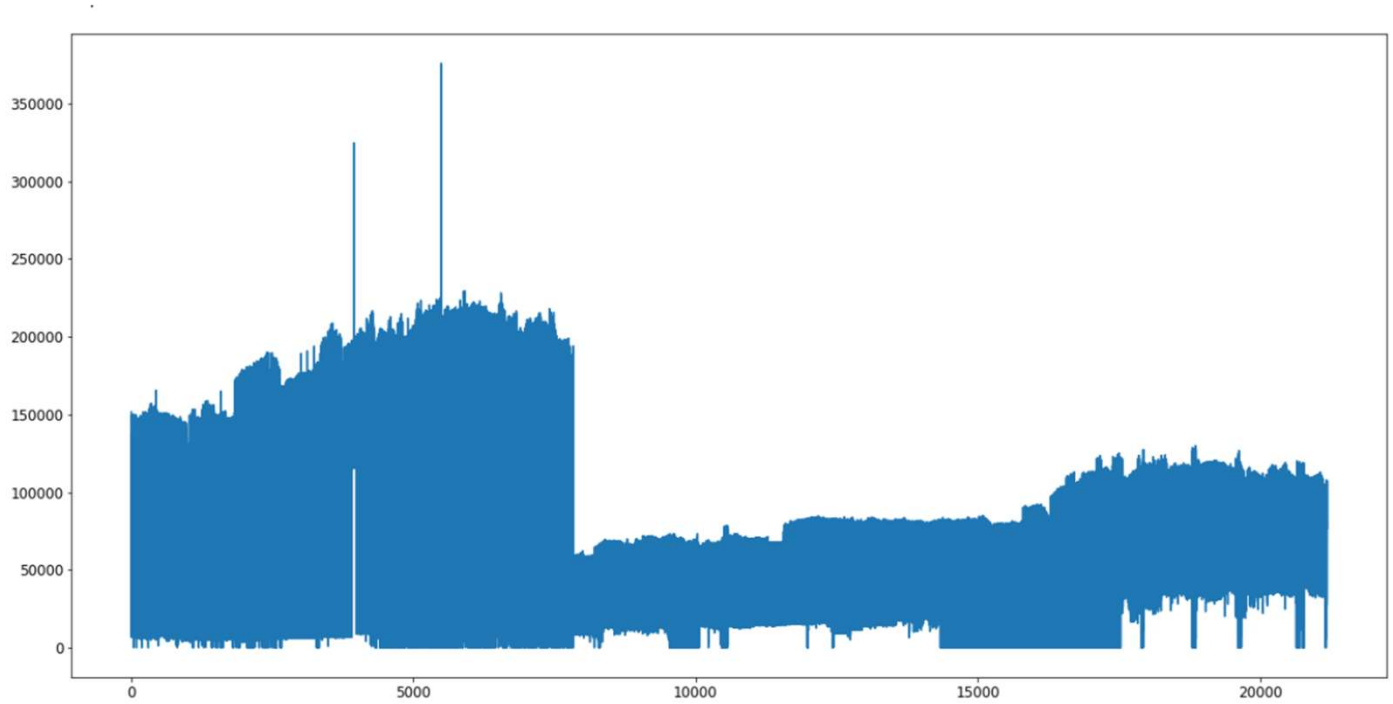
Time series data is a collection of observations obtained through repeated measurements over time. Plot the points on a graph, and one of your axes would always be time.An observed time series can be decomposed into three components: the trend (long term direction), the seasonal (systematic, calendar related movements) and the irregular (unsystematic, short term fluctuations).

A seasonal effect is a systematic and calendar related effect. Some examples include the sharp escalation in most Retail series which occurs around December in response to the Christmas period, or an increase in water consumption in summer due to warmer weather.

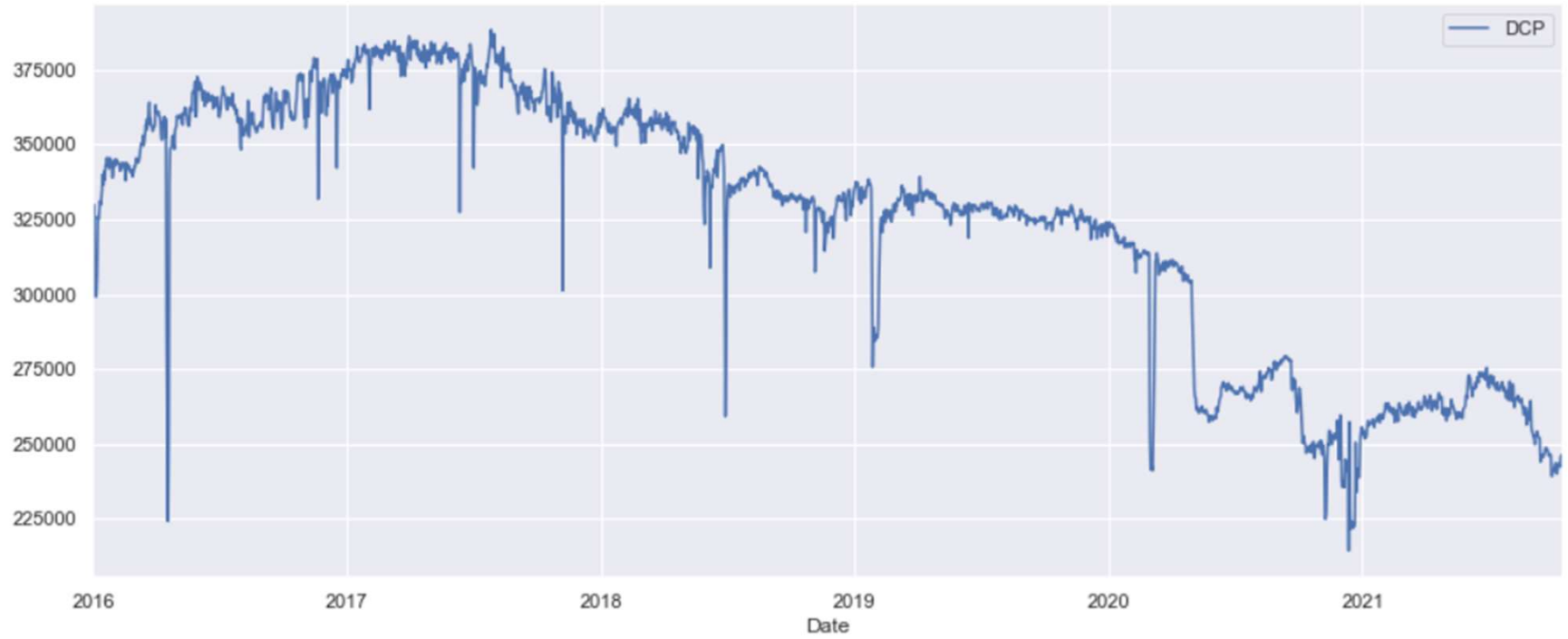
Time series analysis typically requires a large number of data points to ensure consistency and reliability. An extensive data set ensures you have a representative sample size and that analysis can cut through noisy data. It also ensures that any trends or patterns discovered are not outliers and can account for seasonal variance.

I)Data preprocessing and cleaning

Plotting the graph before processing the data



Aggregating the data according to individual GC and dropping the null values



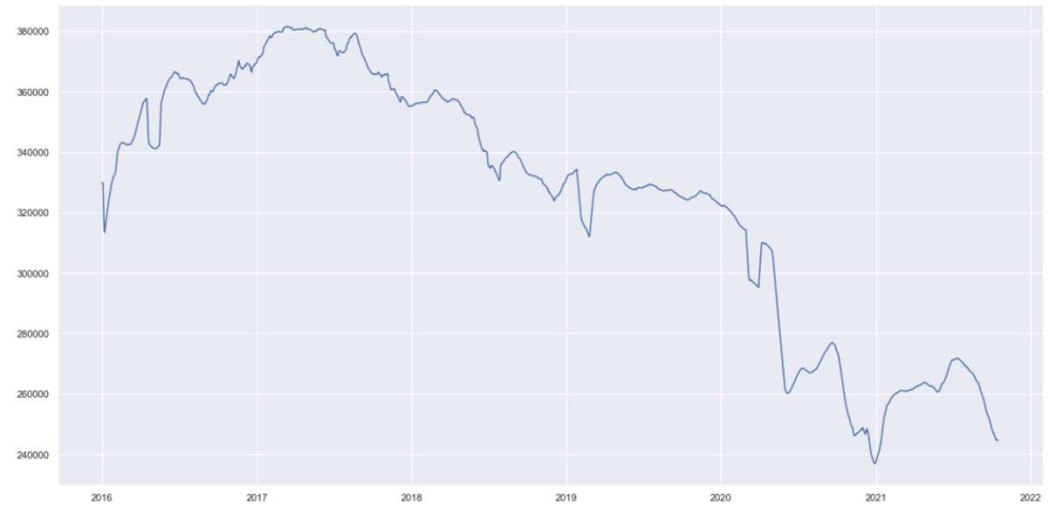
II)Test of Stationarity

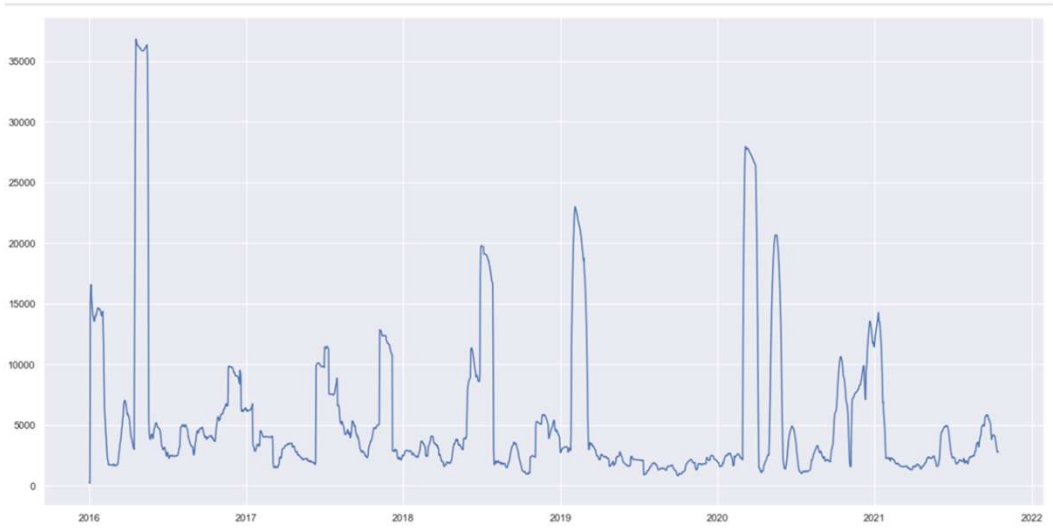
After preprocessing the data, we need to check the stationarity of the data. A dataset is said to be stationary if the mean, variance and autocorrelation remain constant. To check the stationarity, there are a few tests to apply.

Rolling mean/ moving average : A moving average is a technique to get an overall idea of the trends in a data set; it is a calculation to analyze data points by creating a series of averages of different subsets of the full data set.

#Test1 - Rolling statistics

For this test, we plot the rolling mean and rolling variance of the data and observe whether the visualisation is constant in its mean and variance. Both variance and mean have to be constant for this test to pass.





As observed above, the rolling mean is not constant as it is following the downward trend. The rolling variance is a relatively constant plot. Therefore, the data does not pass the first stationarity test.

#Test2 - Augmented Dickey Fuller Test (ADF test)

```
Observations of Dickey-fuller test
Test Statistic                  -0.221899
p-value                          0.935861
#lags used                      21.000000
number of observations used    2094.000000
critical value (1%)              -3.433477
critical value (5%)              -2.862921
critical value (10%)             -2.567505
dtype: float64
```

If p-value is very less than the significance level of 0.05 and we can reject the null hypothesis and consider the series is stationary. But in our case the p-value is 0.935 > 0.05 and hence we cannot reject the null hypothesis and the series is not stationary

#Test3- KPSS (Kwiatkowski-Phillips-Schmidt-Shin)

```
KPSS Statistic: 0.9925455741081595
p-value: 0.01
num lags: 28
Critial Values:
   10% : 0.119
   5% : 0.146
   2.5% : 0.176
   1% : 0.216
Result: The series is not stationary
```

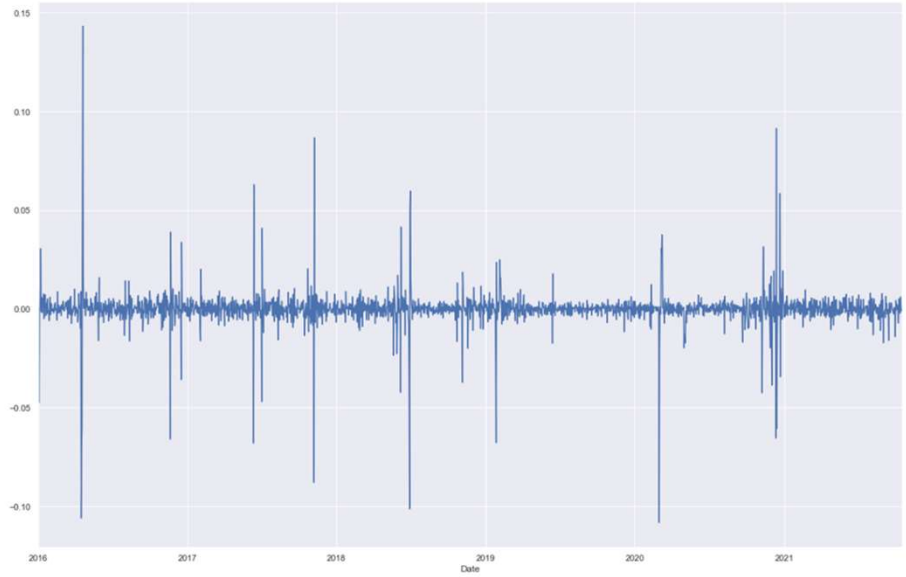
In this test, if p-value is < signif level (say 0.05), then the series is non-stationary. We can observe that the p-value is 0.01 < 0.05 and therefore the data is not stationary

III)Converting the series into stationary series

Differencing is computing the differences between consecutive observations. This is one of the methods to convert non-stationary data into stationary.

Plotting after the differencing of the data. It can be observed that the visual representation demonstrates a stationary dataset

By applying the 3 test of stationarity on the modified dataset, it can be concluded that the series is now stationary



IV)Decomposition of data

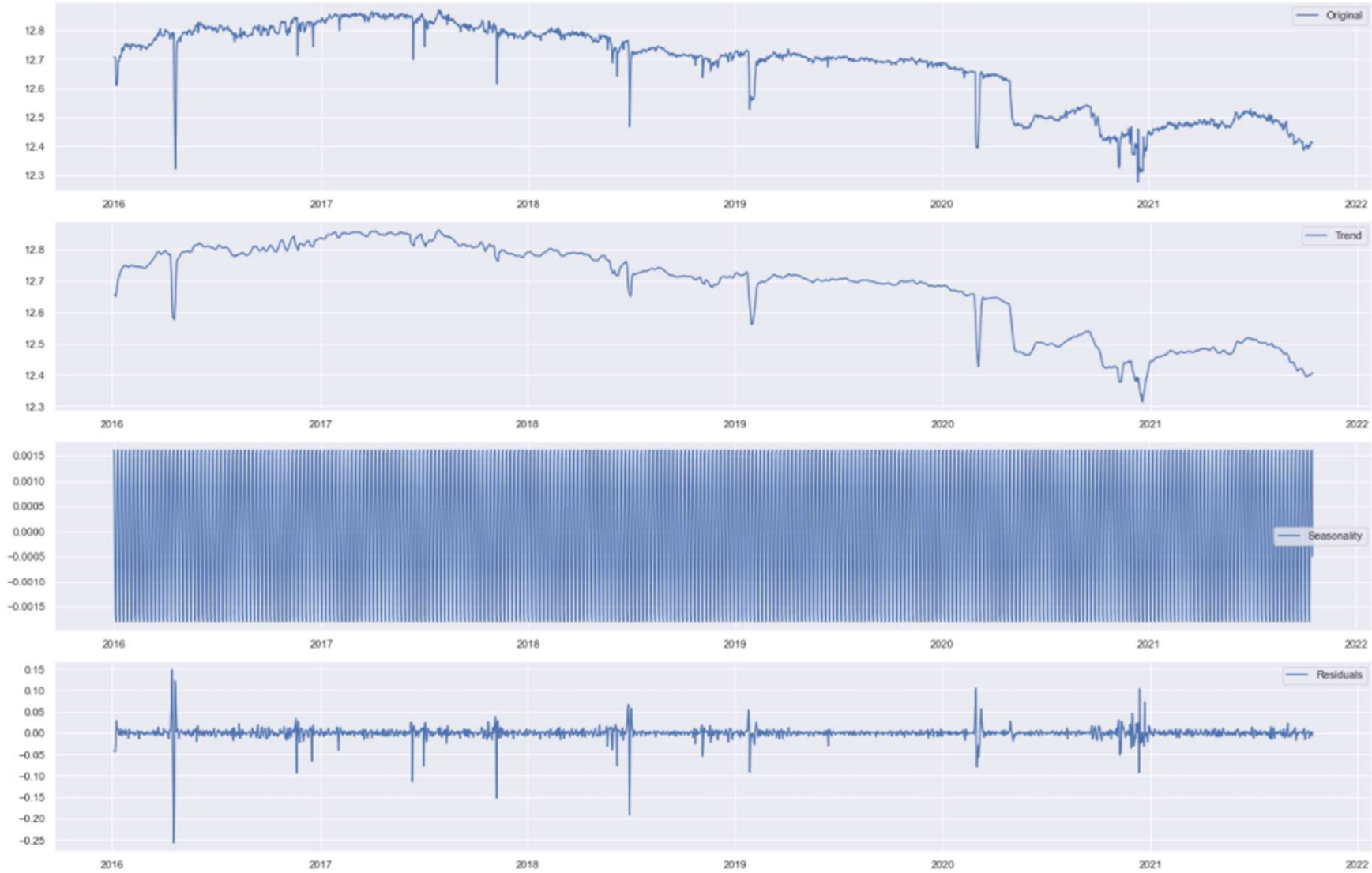
A time series has 4 main components namely, seasonality, trend, cyclicity and residuals.

Trend component: The long-term pattern shown by a time series is called as trend. A trend can be positive or negative depending on whether the time series exhibits an increasing long-term pattern or a decreasing long-term pattern. In other words, we can say a time series has upward trend if it has an overall growth and for negative trend, it shows overall negative growth. If a time series does not show an increasing or decreasing pattern then the series is stationary in the mean.

Seasonal component: Seasonality occurs when the time series exhibits regular fluctuations during the same month (or months) every year, or during the same quarter every year. For instance, in case of sales data of car companies, usually the car companies give a huge discount in the month of December, so there sales increase in the month of December. This happens every year.

Cyclical component: We say the data has a cyclic component when it exhibits rises and falls that are not of fixed period. The duration of a cycle depends on the type of business or industry being analysed.

Irregular component: This component is unpredictable. Every time series has some unpredictable component that makes it a random variable.


Original
12.8
12.7
12.6
12.5
12.4
12.3
2016
2017
2018
2019
2020
2021
2022
Trend
12.8
12.7
12.6
12.5
12.4
12.3
2016
2017
2018
2019
2020
2021
2022
0.0015
0.0010
0.0005
0.0000
−0.0005
−0.0010
−0.0015
Seasonality
2016
2017
2018
2019
2020
2021
2022
0.15
0.10
0.05
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
Residuals
2016
2017
2018
2019
2020
2021
2022

V) Statistical Models

1)Autoregressive Integrated Moving Average (ARIMA)

In an ARIMA model there are 3 parameters that are used to help model the major aspects of a times series: seasonality, trend, and noise. These parameters are labeled p,d,and q.

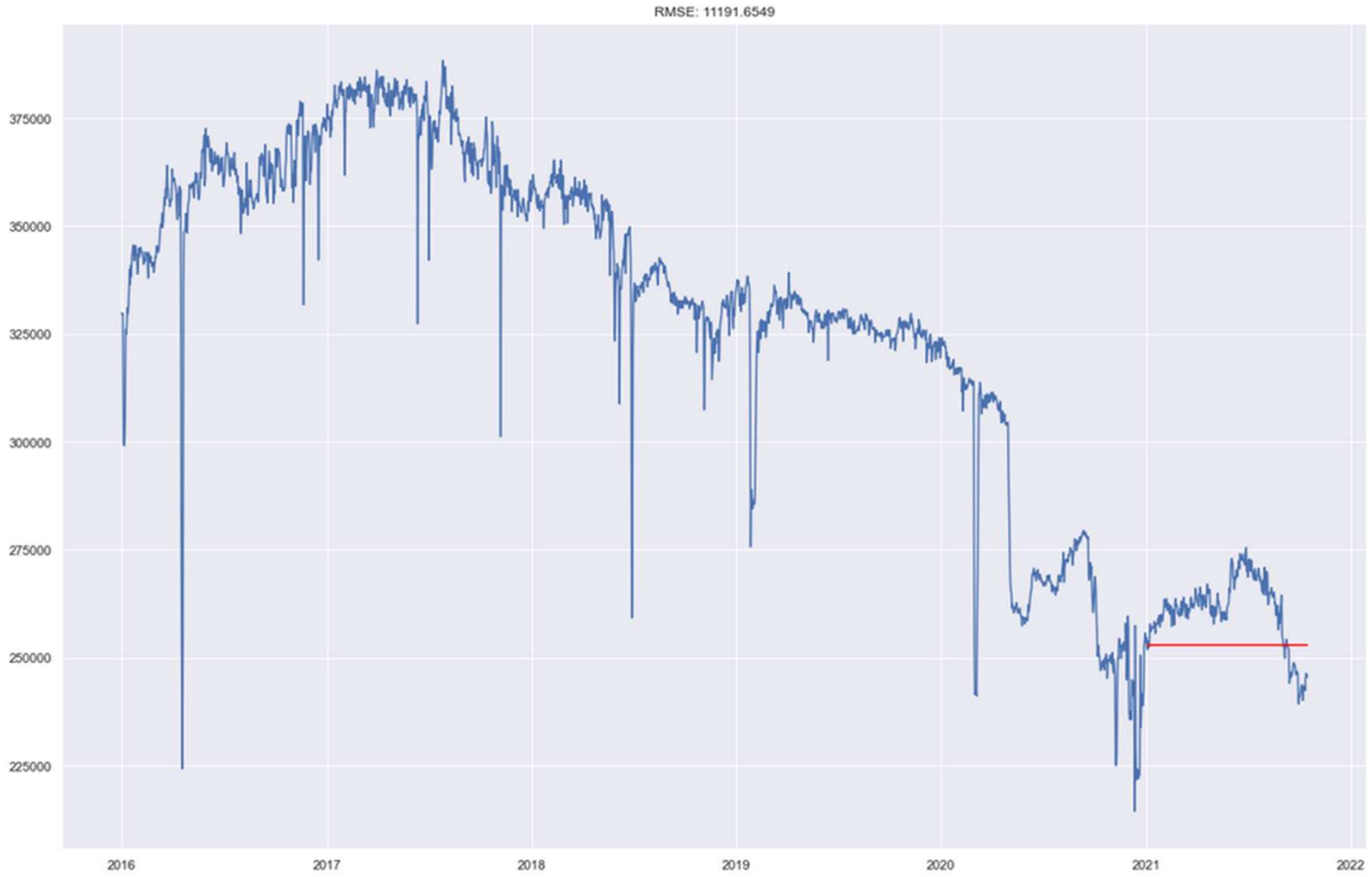
- **Number of AR (Auto-Regressive) terms (p):** p is the parameter associated with the auto-regressive aspect of the model, which incorporates past values i.e lags of dependent variable.
- **Number of Differences (d):** d is the parameter associated with the integrated part of the model, which effects the amount of **differencing** to apply to a time series.
- **Number of MA (Moving Average) terms (q):** q is size of the moving average part window of the model i.e. lagged forecast errors in prediction equation.

Observations from ARIMA model (6,1,3)

Dep. Variable:	DCP	**No. Observations:**	1830
Model:	ARIMA(6, 1, 3)	**Log Likelihood**	-18504.288
Date:	Sun, 14 Nov 2021	**AIC**	37028.576
Time:	12:55:04	**BIC**	37083.691
Sample:	01-01-2016	**HQIC**	37048.905
	- 01-03-2021		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-0.0234	5.375	-0.004	0.997	-10.558	10.511
ar.L2	-0.0239	4.493	-0.005	0.996	-8.829	8.781
ar.L3	-0.0510	1.749	-0.029	0.977	-3.480	3.378
ar.L4	-0.0486	0.422	-0.115	0.908	-0.875	0.778
ar.L5	-0.0267	0.221	-0.121	0.904	-0.459	0.406
ar.L6	-0.0081	0.058	-0.140	0.888	-0.121	0.105
ma.L1	-0.0188	5.374	-0.003	0.997	-10.551	10.514
ma.L2	-0.0212	4.676	-0.005	0.996	-9.186	9.144
ma.L3	-0.0513	1.810	-0.028	0.977	-3.599	3.497
sigma2	3.478e+07	2.35e-05	1.48e+12	0.000	3.48e+07	3.48e+07

Ljung-Box (L1) (Q):	2.10	**Jarque-Bera (JB):**	227082.45
Prob(Q):	0.15	**Prob(JB):**	0.00
Heteroskedasticity (H):	0.45	**Skew:**	-1.53
Prob(H) (two-sided):	0.00	**Kurtosis:**	57.50



Inference: It can be observed that the predictions and the actual values are not in alignment due to the convergence of the time series to the mean. Therefore this model does not give a very accurate predictions of the DCP value

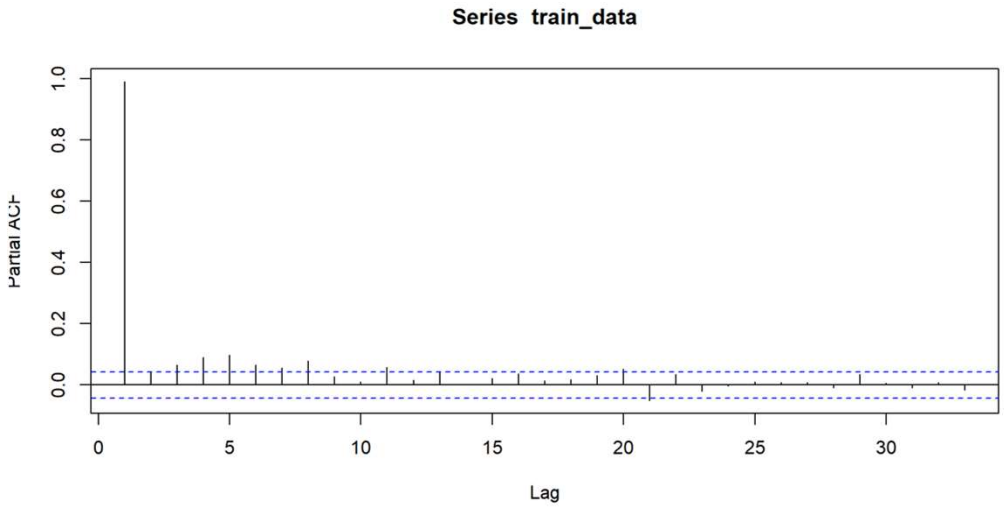
2)Seasonal Autoregressive Integrated Moving-Average (SARIMA)

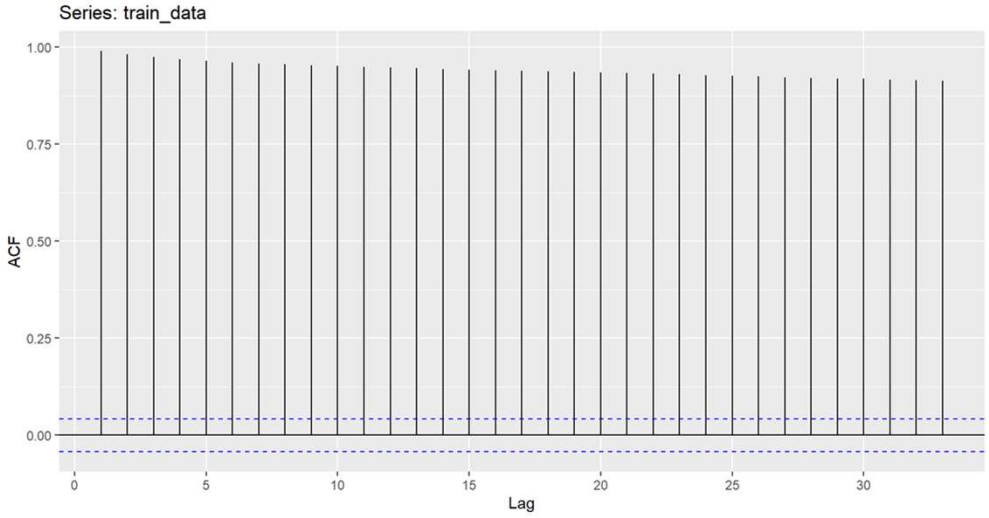
It is an extension of ARIMA that explicitly supports univariate time series data with a seasonal component.This model finds the optimal value for p,q,d for all different combinations and calculates the Akaike's Information Criterion (AIC). The combination with the lowest AIC is chosen as the optimal value. In the above figure, the lowest AIC= 42531 is for the model with values (3,1,1)

```
Performing stepwise search to minimize aic
 ARIMA(2,1,2)(0,0,0)[0] intercept   : AIC=42547.053, Time=1.54 sec
 ARIMA(0,1,0)(0,0,0)[0] intercept   : AIC=42599.144, Time=0.04 sec
 ARIMA(1,1,0)(0,0,0)[0] intercept   : AIC=42598.601, Time=0.09 sec
 ARIMA(0,1,1)(0,0,0)[0] intercept   : AIC=42598.641, Time=0.10 sec
 ARIMA(0,1,0)(0,0,0)[0]             : AIC=42597.247, Time=0.03 sec
 ARIMA(1,1,2)(0,0,0)[0] intercept   : AIC=42556.371, Time=0.93 sec
 ARIMA(2,1,1)(0,0,0)[0] intercept   : AIC=42553.497, Time=0.50 sec
 ARIMA(3,1,2)(0,0,0)[0] intercept   : AIC=42536.250, Time=2.56 sec
 ARIMA(3,1,1)(0,0,0)[0] intercept   : AIC=42533.599, Time=0.87 sec
 ARIMA(3,1,0)(0,0,0)[0] intercept   : AIC=42549.698, Time=0.24 sec
 ARIMA(4,1,1)(0,0,0)[0] intercept   : AIC=42535.719, Time=1.15 sec
 ARIMA(2,1,0)(0,0,0)[0] intercept   : AIC=42596.064, Time=0.14 sec
 ARIMA(4,1,0)(0,0,0)[0] intercept   : AIC=42539.508, Time=0.33 sec
 ARIMA(4,1,2)(0,0,0)[0] intercept   : AIC=42537.983, Time=1.25 sec
 ARIMA(3,1,1)(0,0,0)[0]             : AIC=42531.345, Time=0.83 sec
 ARIMA(2,1,1)(0,0,0)[0]             : AIC=42550.248, Time=0.41 sec
 ARIMA(3,1,0)(0,0,0)[0]             : AIC=42547.727, Time=0.22 sec
 ARIMA(4,1,1)(0,0,0)[0]             : AIC=42533.366, Time=1.00 sec
 ARIMA(3,1,2)(0,0,0)[0]             : AIC=42533.709, Time=2.43 sec
 ARIMA(2,1,0)(0,0,0)[0]             : AIC=42594.132, Time=0.11 sec
 ARIMA(2,1,2)(0,0,0)[0]             : AIC=42542.769, Time=1.10 sec
 ARIMA(4,1,0)(0,0,0)[0]             : AIC=42537.500, Time=0.33 sec
 ARIMA(4,1,2)(0,0,0)[0]             : AIC=42535.768, Time=1.08 sec

Best model:  ARIMA(3,1,1)(0,0,0)[0]
Total fit time: 17.314 seconds
```

Predictions of the SARIMA model

AutoCorrelation Function (ACF) Plot and partial autocorrelation function (PACF)

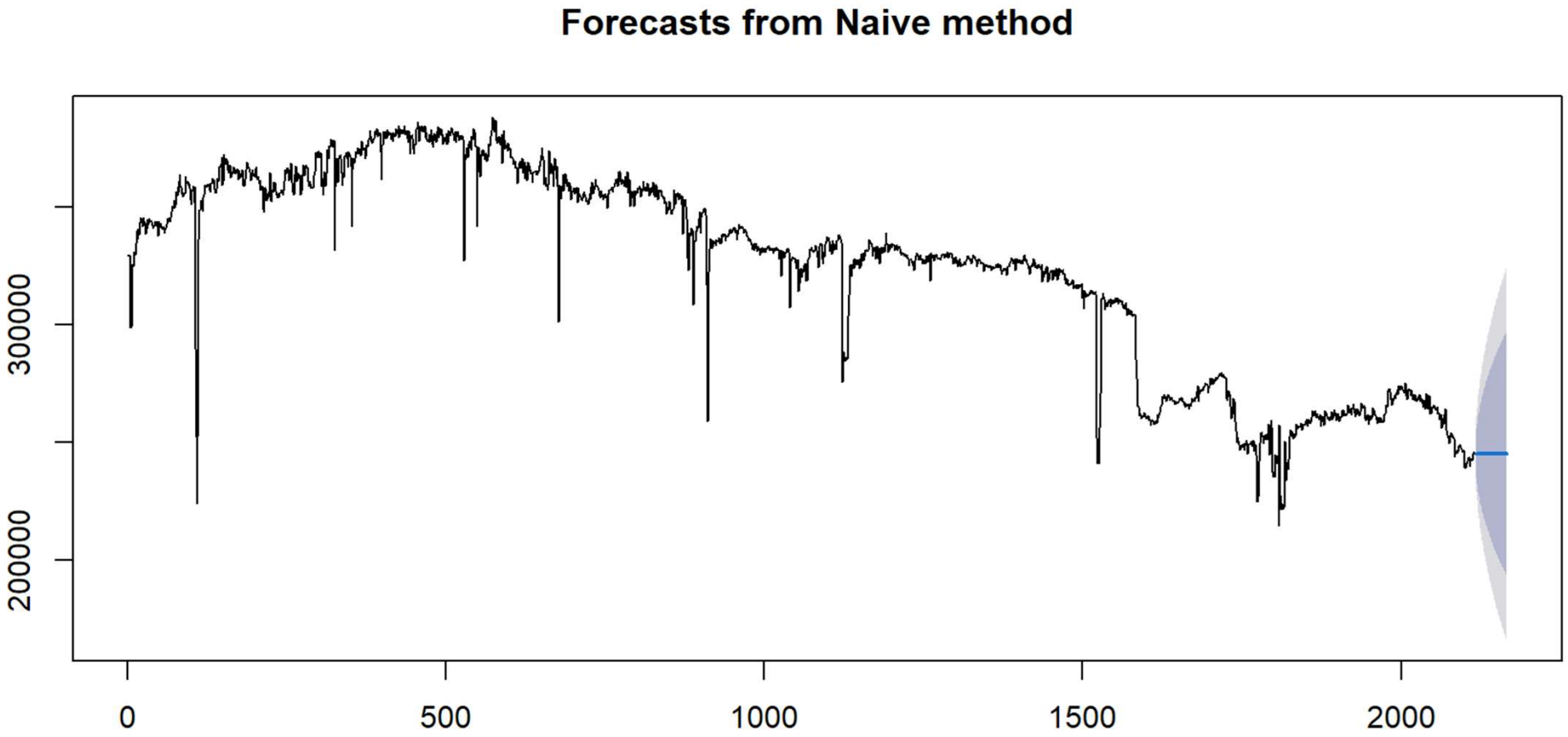




3)Naive bayes method

Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.Bayes' theorem is also known as Bayes' Rule or Bayes' law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

```
Forecast method: Naive method

Model Information:
Call: naive(y = train_data, h = 50)

Residual sd: 5715.0588

Error measures:
                     ME     RMSE      MAE         MPE
Training set -39.92719 5715.059 2434.679 -0.0319393
                   MAPE MASE         ACF1
Training set 0.7715194    1 -0.05310757

Forecasts:
     Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
2117         245421 238096.9 252745.1 234219.7 256622.3
2118         245421 235063.1 255778.9 229580.0 261262.0
2119         245421 232735.2 258106.8 226019.8 264822.2
2120         245421 230772.7 260069.3 223018.4 267823.6
2121         245421 229043.7 261798.3 220374.1 270467.9
2122         245421 227480.6 263361.4 217983.5 272858.5
2123         245421 226043.1 264798.9 215785.1 275056.9
```


Forecasts from Naive method
300000
200000
0
500
1000
1500
2000

4)Holts- winter model

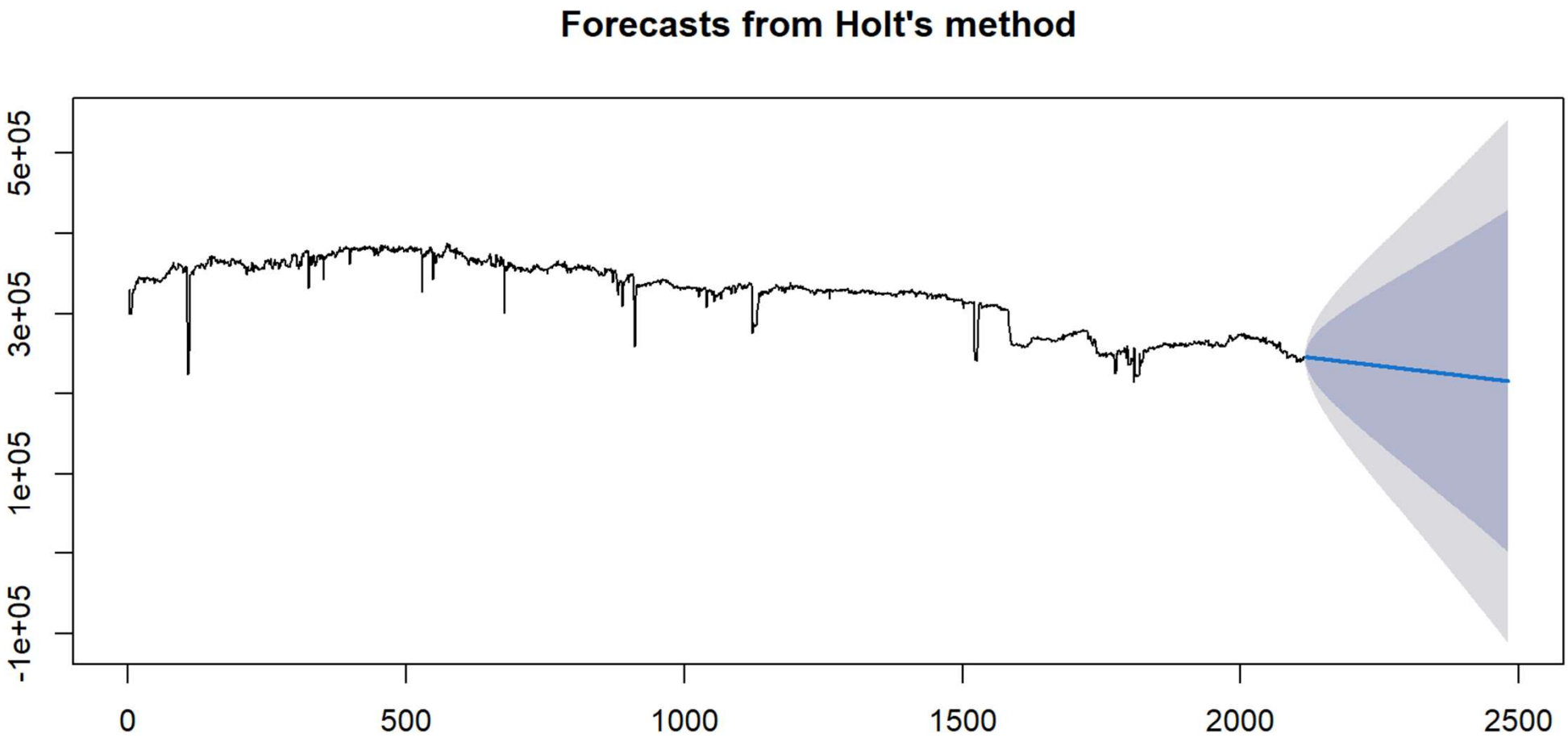
Holt-Winters is a model of time series behavior. Holt-Winters is a way to model three aspects of the time series: a typical value (average), a slope (trend) over time, and a cyclical repeating pattern (seasonality).The Holt-Winters method uses exponential smoothing to encode lots of values from the past and use them to predict “typical” values for the present and future.

Observations from holt's method

```
Forecast method: Holt's method

Model Information:
Holt's method

Call:
 holt(y = train_data, h = 100)

  Smoothing parameters:
    alpha = 0.9395
    beta  = 0.003

  Initial states:
    l = 318976.283
    b = 1087.6815

  sigma:  5729.707

     AIC     AICc      BIC
52830.08 52830.11 52858.36
```

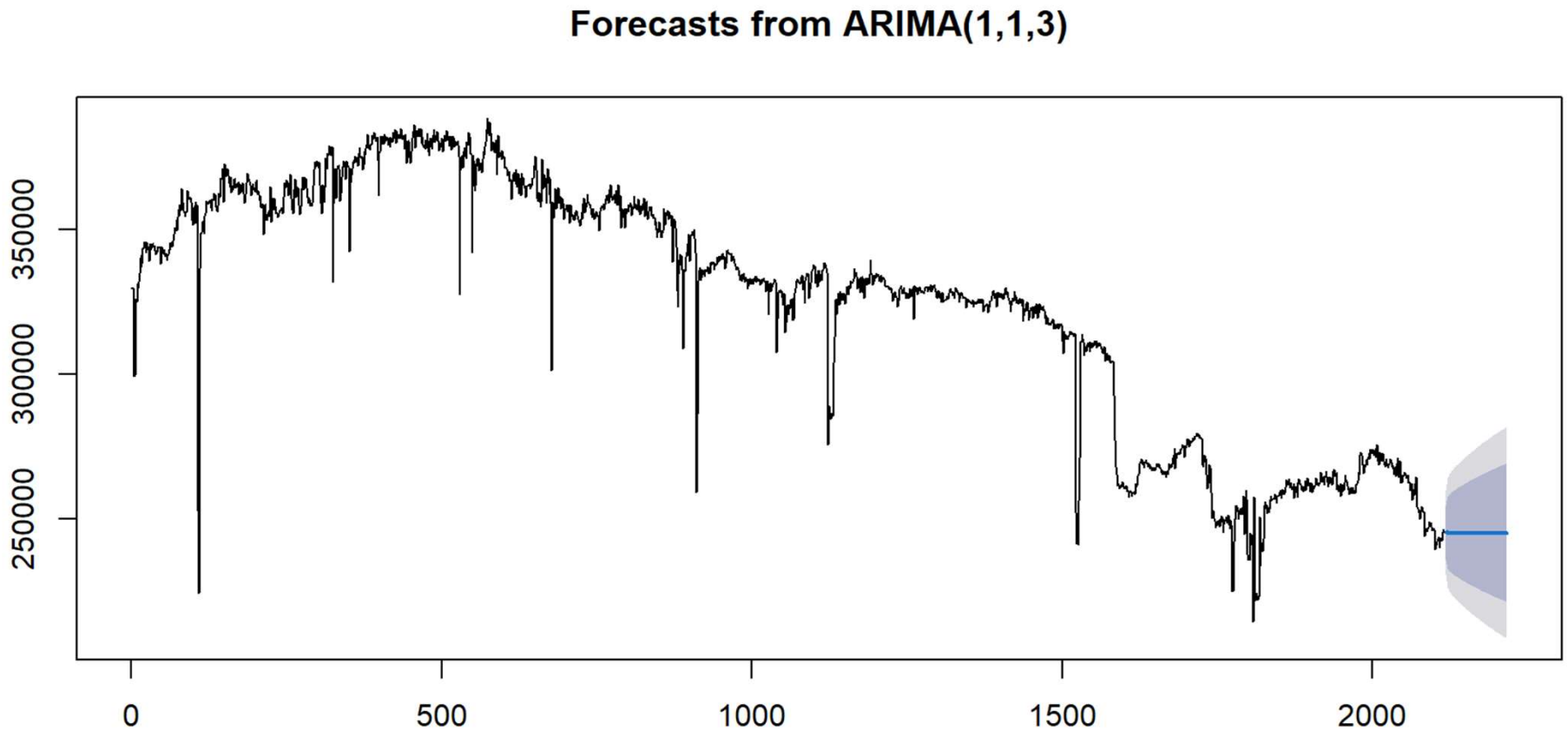
Error measures and forecast of holt's method

```
Error measures:
                      ME     RMSE      MAE          MPE
Training set -182.3825 5724.288 2452.062 -0.06949238
                    MAPE    MASE         ACF1
Training set 0.7770114 1.00714 0.003931777

Forecasts:
     Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
2117       245370.0 238027.1 252712.9 234140.0 256600.0
2118       245287.6 235197.0 255378.1 229855.4 260719.7
2119       245205.1 232956.5 257453.7 226472.5 263937.7
2120       245122.7 231032.0 259213.4 223572.9 266672.5
2121       245040.3 229312.0 260768.5 220986.0 269094.5
2122       244957.8 227738.1 262177.5 218622.6 271293.1
2123       244875.4 226275.1 263475.7 216428.6 273322.1
2124       244792.9 224899.7 264686.2 214368.9 275217.0
2125       244710.5 223596.0 265825.0 212418.7 277002.3
2126       244628.1 222352.1 266904.1 210559.9 278696.3
2127       244545.6 221159.0 267932.3 208778.9 280312.4
```


Forecasts from Holt's method
5e+05
3e+05
1e+05
-1e+05
0
500
1000
1500
2000
2500

5)ARIMA (1,1,3)

```
Series: train_data
ARIMA(1,1,3)

Coefficients:
         ar1      ma1      ma2      ma3
      0.6957  -0.8140  -0.0426  -0.0550
s.e.  0.0452   0.0489   0.0293   0.0295

sigma^2 = 30609615:  log likelihood = -21227.23
AIC=42464.45   AICc=42464.48   BIC=42492.73

Training set error measures:
                     ME     RMSE      MAE          MPE
Training set -131.7713 5526.055 2488.656 -0.07581036
                   MAPE    MASE          ACF1
Training set 0.7965411 1.02217 0.0001753418
```


Forecasts from ARIMA(1,1,3)
250000
300000
350000
0
500
1000
1500
2000

6) Double-Seasonal Holt-Winters Forecasting

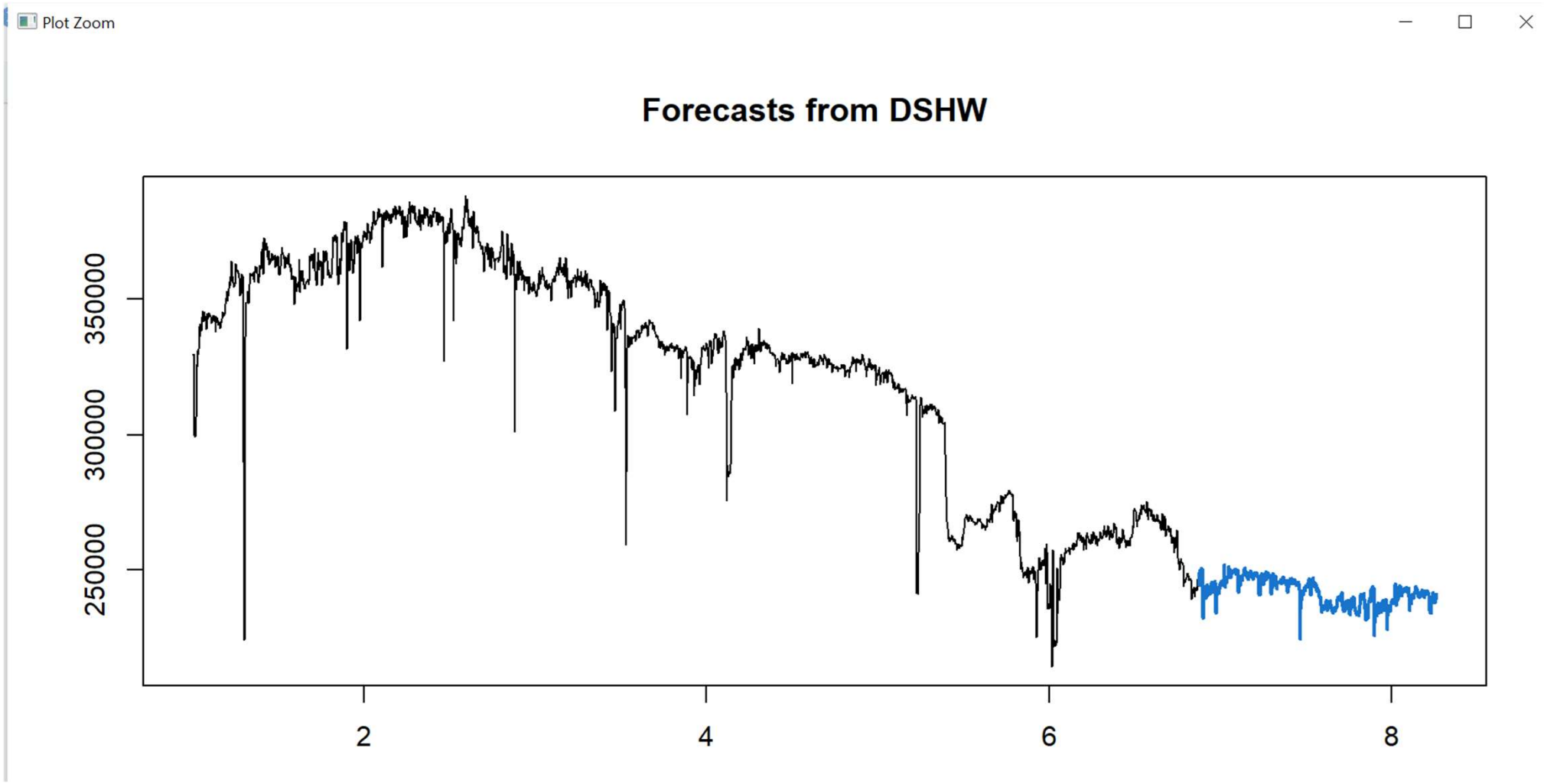
This method uses additive trend and multiplicative seasonality, where there are two seasonal components which are multiplied together.

Seasonal component 1 - period1=12

Seasonal component 2 - period2=360

The model essentially enables it to capture two seasonal cycles of the data.a smaller one repeated often and a bigger one repeated less often. For the method to work however, the seasonalities need to be nested, meaning one must be an integer multiple of the other.

```
Error measures:
                      ME      RMSE       MAE          MPE
Training set -112.9267 6344.027 3022.212 -0.05667543
                    MAPE      MASE       ACF1
Training set 0.971131 0.1038972 0.09583052

Forecasts:
       Point Forecast
6.8778       244176.9
6.8806       249181.7
6.8833       249603.3
6.8861       248005.1
6.8889       250482.8
6.8917       249275.9
6.8944       248811.2
6.8972       249930.9
6.9000       231971.9
6.9028       236855.2
```


Plot Zoom
Forecasts from DSHW
250000
300000
350000
2
4
6
8

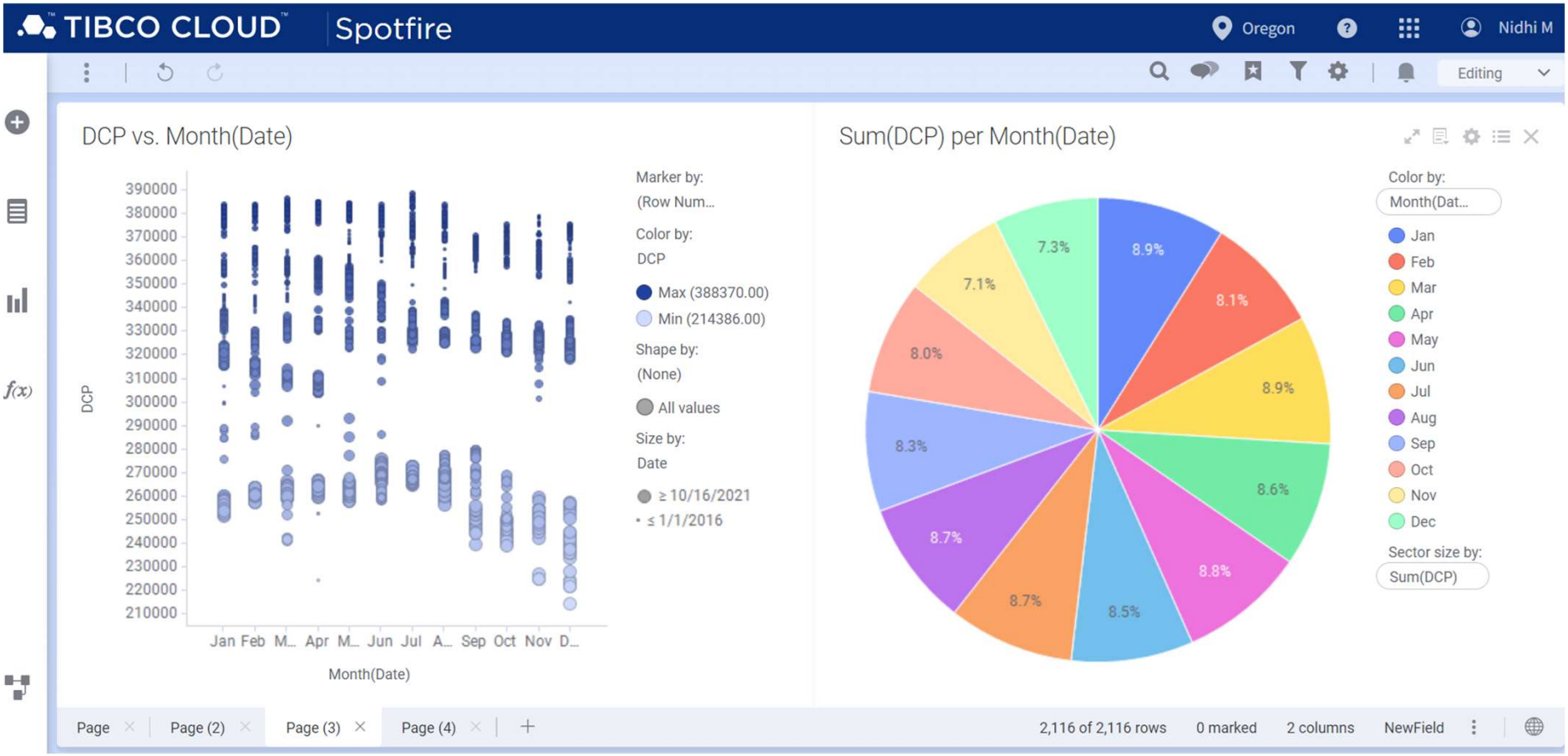
RESULTS

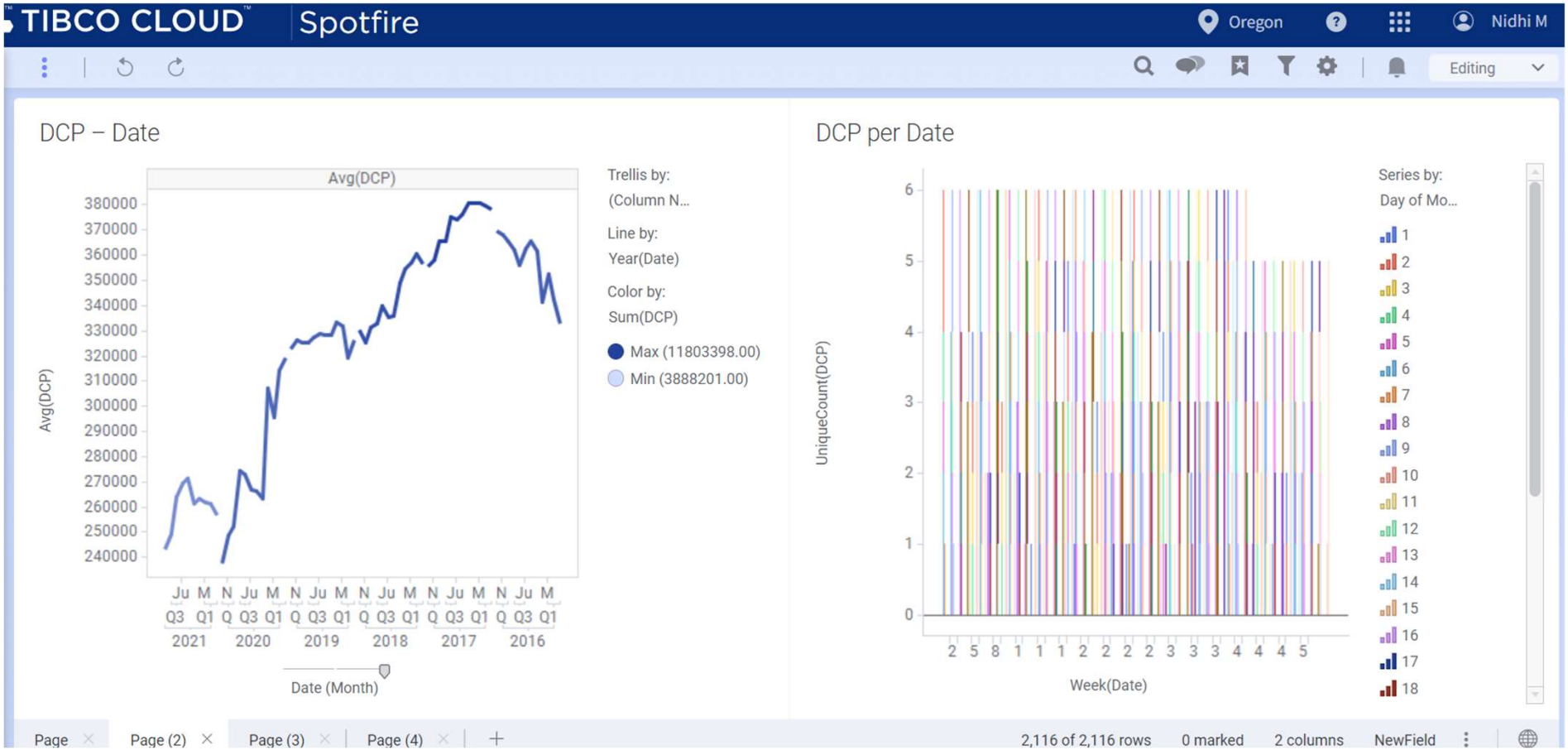
Statistical/ machine learning models used:

1) ARIMA (6,1,3)
2) SARIMA
3) Naive bayes
4) Holts-winter
5) ARIMA (1,1,3)
6) Double seasonal holts winter (DSHW)

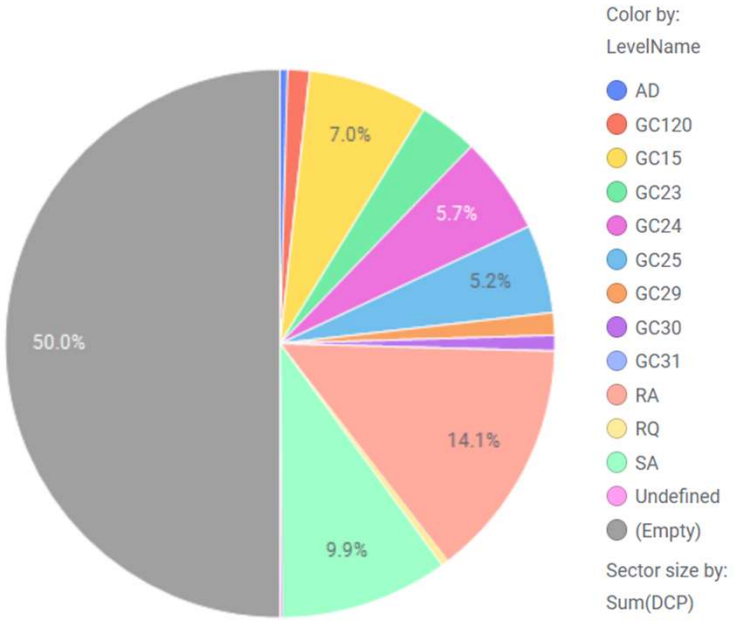
On comparing the results of different models, it can be inferred that the double seasonal holts winter model gave the best predictions of the DCP values.

Data Visualization using spotfire

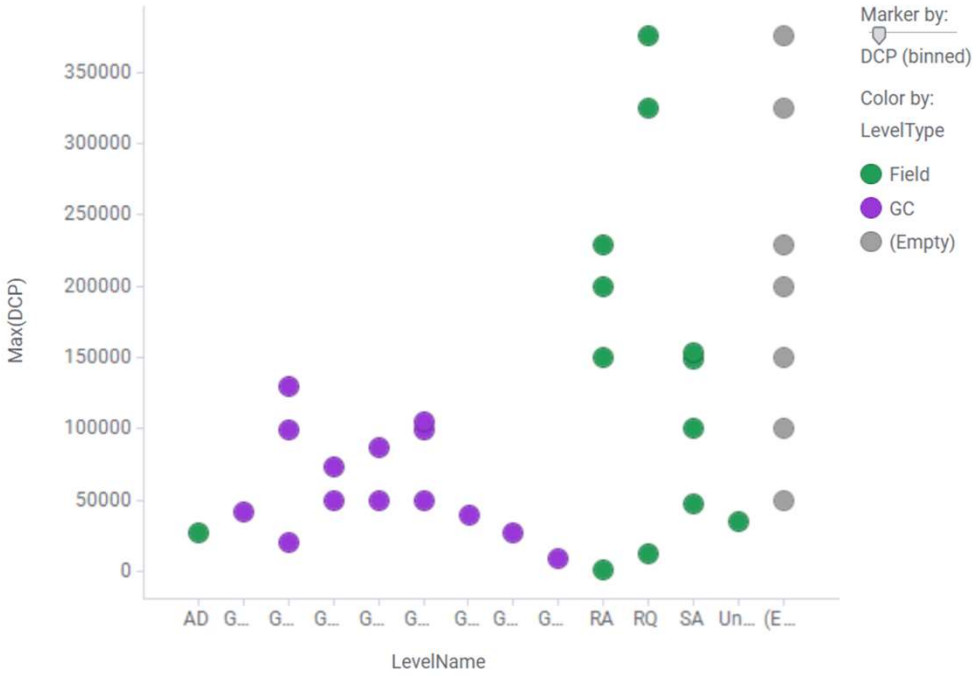



TIBCO CLOUD
Spotfire
Oregon
Nidhi M
Editing
DCP – Date
Avg(DCP)
Avg(DCP)
Date (Month)
Trellis by:
(Column N...
Line by:
Year(Date)
Color by:
Sum(DCP)
Max (11803398.00)
Min (3888201.00)
DCP per Date
UniqueCount(DCP)
Week(Date)
Series by:
Day of Mo...
Page
Page (2)
Page (3)
Page (4)
2,116 of 2,116 rows
0 marked
2 columns
NewField

Sum(DCP) per LevelName



Max(DCP) vs. LevelName

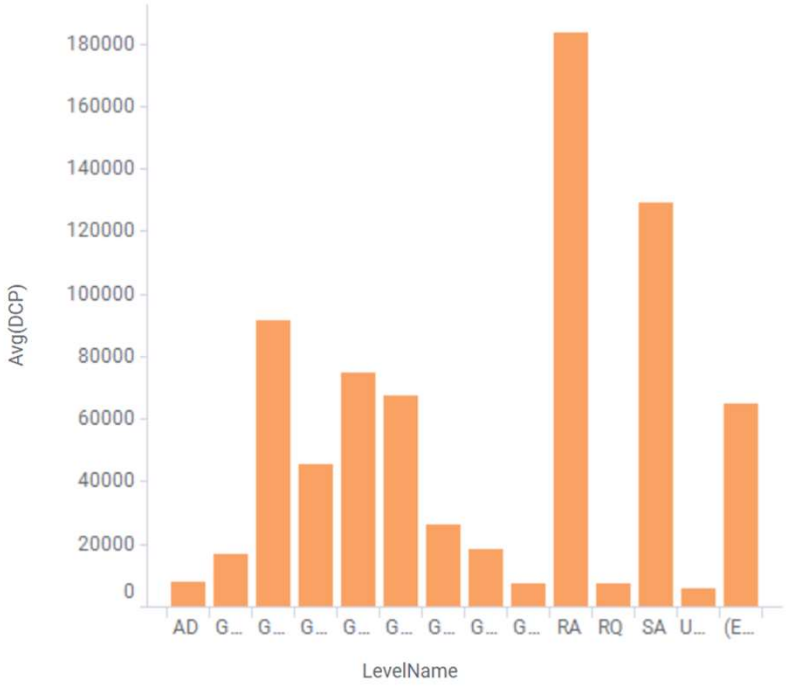


TIBCO CLOUD™ Spotfire

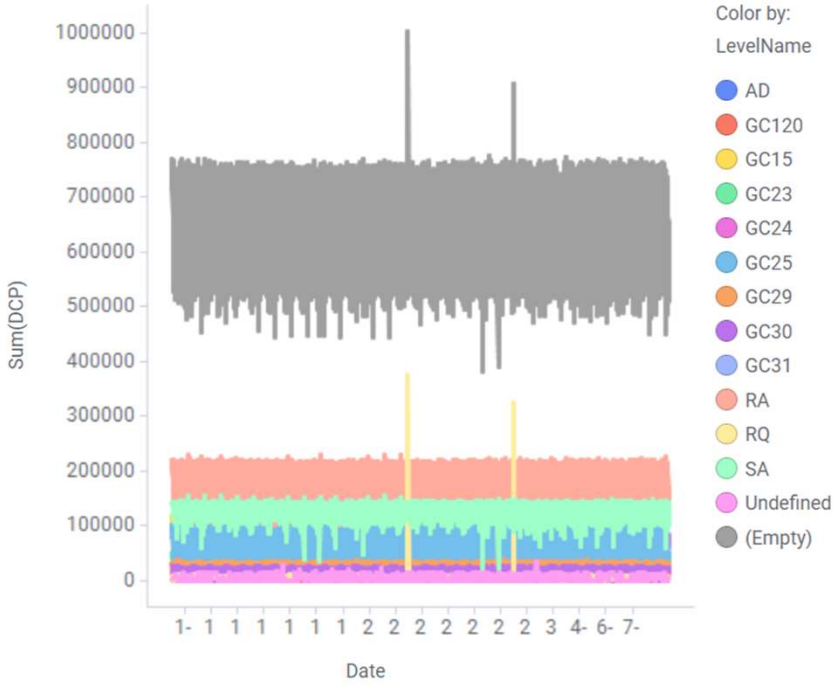
Oregon | Nidhi M

File Edit Data Visualizations View Tools Help User | Viewing

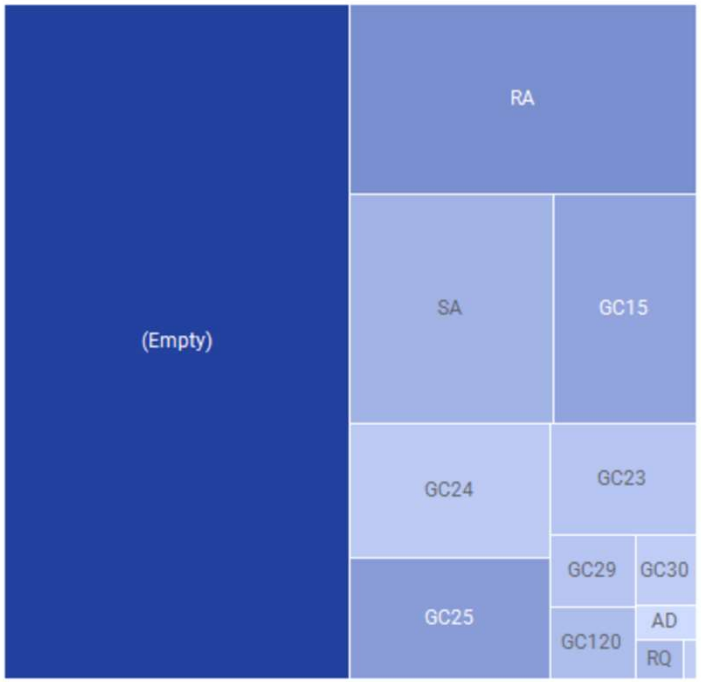
DCP per LevelName



DCP – Date



DCP per LevelName



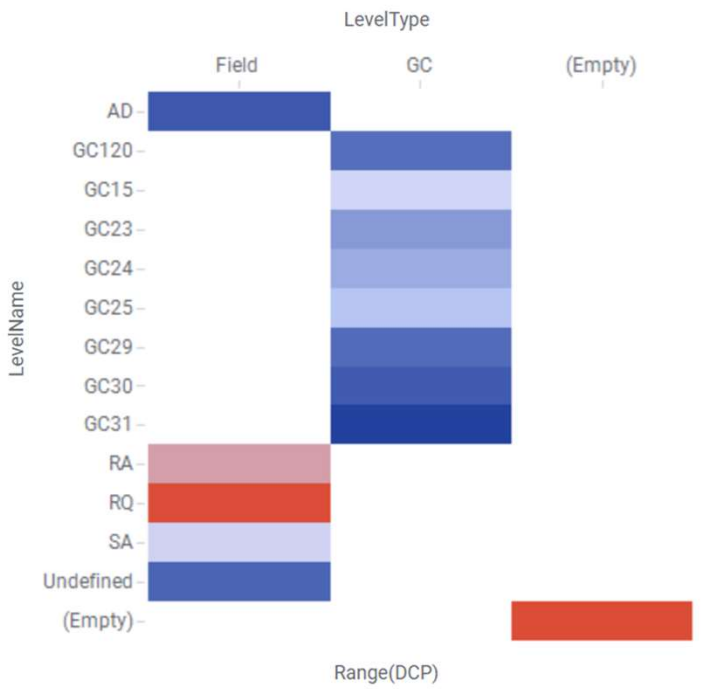
Color by:
StdDev(DCP)

- Max (56879.77)
- Min (1880.16)

Size by:
Sum(DCP)

Hierarchy:
LevelName

DCP per LevelType and LevelName



Colors:

- Max
- Average
- Min