

# SARIMA

Songül Karakaya

2025-06-10

## Contents

<b>Zaman Serisinin Tanıtılması ve Duraganlık Kontrolleri</b>	<b>2</b>
Serinin Görselleştirilmesi ve Dekompozisyon . . . . .	2
Orijinal Serinin Duraganlık Analizi . . . . .	3
<b>Orijinal Seride Fark Alma</b>	<b>5</b>
Birinci Derece Normal Fark (d=1) . . . . .	5
Birinci Derece ve Mevsimsel Fark (d=1, D=1) . . . . .	6
<b>Serinin Logaritmik Dönüşümü ve Fark Alma</b>	<b>8</b>
Logaritmik Dönüşüm . . . . .	8
Log Dönüşümü + Birinci Fark (d=1) . . . . .	9
Log Dönüşümü + Birinci ve Mevsimsel Fark (d=1, D=1) . . . . .	10
İkinci Dereceden Fark Alma İşlemleri . . . . .	11
<b>Model Belirleme</b>	<b>14</b>
Otomatik Model Belirleme ( <code>auto.arima</code> ) . . . . .	14
Manuel Model Denemeleri ve Karşılaştırma . . . . .	16
<b>En Uygun Modele Karar Verme</b>	<b>18</b>
<b>Residual'ların Analizi (Diagnostic)</b>	<b>19</b>
Sonuçların Yorumlanması . . . . .	19
<b>Model ile Gelecek Tahmini (Forecast)</b>	<b>20</b>
Grafiğin Yorumlanması . . . . .	20
<b>Sonuç ve Genel Değerlendirme</b>	<b>21</b>

## Zaman Serisinin Tanıtılması ve Duraganlık Kontrolleri

Bu rapor, 1989 yılından itibaren aylık doğalgaz ithalat fiyat verilerinden oluşturulmuş bir zaman serisini gösterir.

Bu raporda amacımız, verinin yapısını anlamak, durağanlığını test etmek, uygun bir SARIMA modeli kurmak ve bu model ile gelecek dönemler için fiyat tahmini yapmaktır.

### Serinin Görselleştirilmesi ve Dekompozisyon

Zaman serisinin grafiğini görelim.

```
# Zaman serisi grafiği
ts.plot(ts_data,
        main = "Doğalgaz Fiyatları Zaman Serisi",
        ylab = "Fiyat",
        xlab = "Zaman")
```

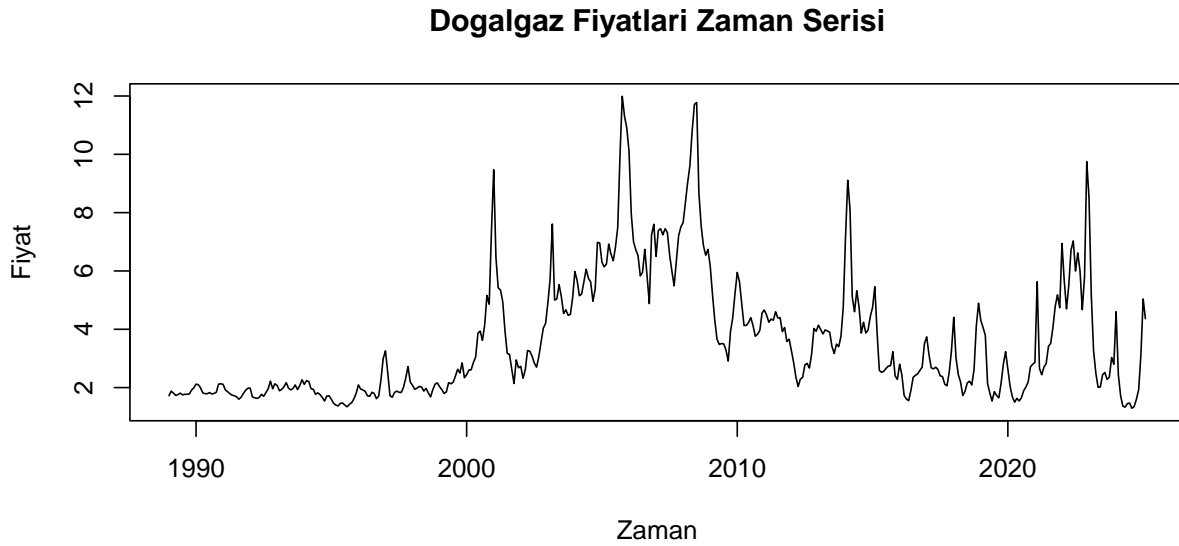


Figure 1: Doğalgaz Fiyatları Zaman Serisi Grafiği (1989-Günümüz)

Seride yukarı yönlü trend ve tekrarlayan mevsimsel dalgalanmalar olduğu görülebilir. Bu yapıları daha net görmek için seriyi bileşenlerine ayıralım. Toplamsal (additive) model, mevsimselliğin boyutunun zamanla değişmediği varsayımına dayanır.

```
# Zaman serisi dekompozisyonu
dekompozisyon <- decompose(ts_data, type = "additive") # (toplamsal model)
plot(dekompozisyon)
```

Bileşenler incelendiğinde serinin **durağan olmadığını** görülmektedir.

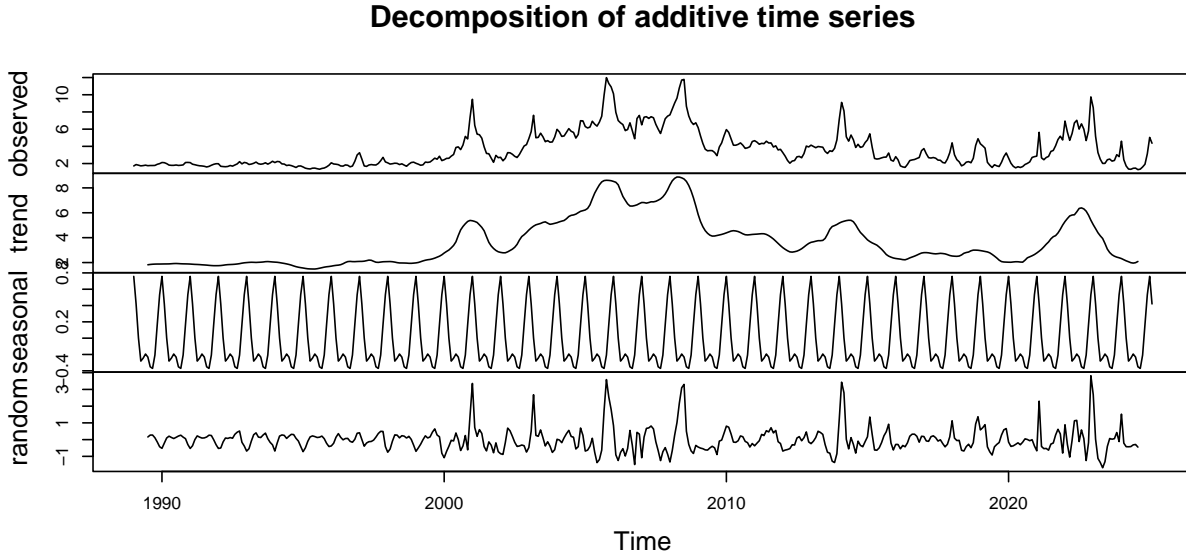


Figure 2: Zaman Serisinin Toplamsal Bileşenlere Ayrıştırılması

## Original Serinin Durağanlık Analizi

Görsel analizde edindiğimiz serinin durağan olmadığı izlenimini istatistiksel testler ile teyit edelim. Bunun için Dickey-Fuller (ADF), Phillips-Perron (PP) ve KPSS testlerini kullanacağız.

```
##
## --- Orijinal Serinin Durağanlık Testleri ---
##
## Augmented Dickey-Fuller Test
##
## data: ts_data
## Dickey-Fuller = -3.2844, Lag order = 7, p-value = 0.07362
## alternative hypothesis: stationary
## Warning in pp.test(ts_data): p-value smaller than printed p-value
##
## Phillips-Perron Unit Root Test
##
## data: ts_data
## Dickey-Fuller Z(alpha) = -28.897, Truncation lag parameter = 5, p-value
## = 0.01
## alternative hypothesis: stationary
## Warning in kpss.test(ts_data, null = "Level"): p-value smaller than printed
## p-value
##
## KPSS Test for Level Stationarity
##
## data: ts_data
## KPSS Level = 1.2587, Truncation lag parameter = 5, p-value = 0.01
```

### Test Sonuçlarının Yorumlanması:

- **ADF Testi:**

p değeri 0.05'ten büyüktür. Bu nedenle,  $H_0$  hipotezini (durağanlık) reddedemeyiz.

- **PP Testi:**

p değeri 0.05'ten küçüktür.  $H_0$  reddedilir, serinin durağan olduğunu söyler.

- **KPSS Testi:**

p-değeri 0.05'ten küçük olduğu için, serinin durağan olduğunu reddederiz.

ADF testi ve KPSS testi serinin durağan olmadığını söylerken PP testi serinin durağan olduğunu söylüyor. Daha net bir sonuca varmak için serinin ACF ve PACF grafiklerine bakalım.

```
par(mfrow=c(1,2)) # Grafikleri 1 satır 2 sütun şeklinde yan yana göstermek için
acf(ts_data, lag.max = 50, main = "Orijinal Seri ACF")
pacf(ts_data, lag.max = 50, main = "Orijinal Seri PACF")
```

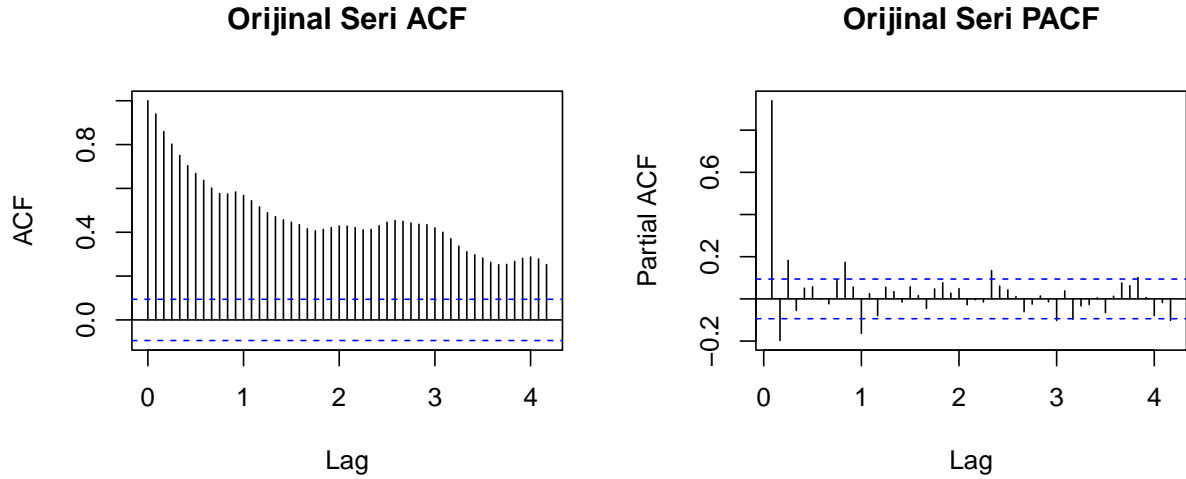


Figure 3: Orijinal Serinin ACF ve PACF Grafikleri

```
par(mfrow=c(1,1)) # Grafik düzenini sıfırlama
```

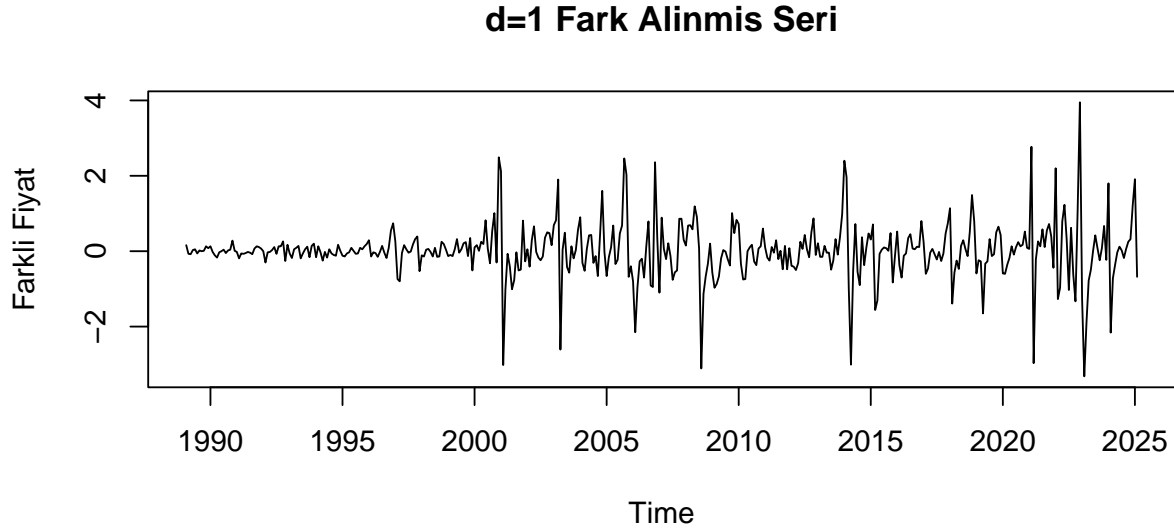
### Grafiklerin Yorumlanması:

ACF grafiğindeki otokorelasyon değerlerinin çok yavaş bir şekilde azalıyor muhtemelen bu durum seride çok güçlü bir mevsimsellik olmasından kaynaklanıyor.

Korelasyon grafikleri, modelleme öncesinde serinin durağanlaştırılması gerektiğini kesin olarak ortaya koyuyor. Durağanlaştırmak için fark alacağız.

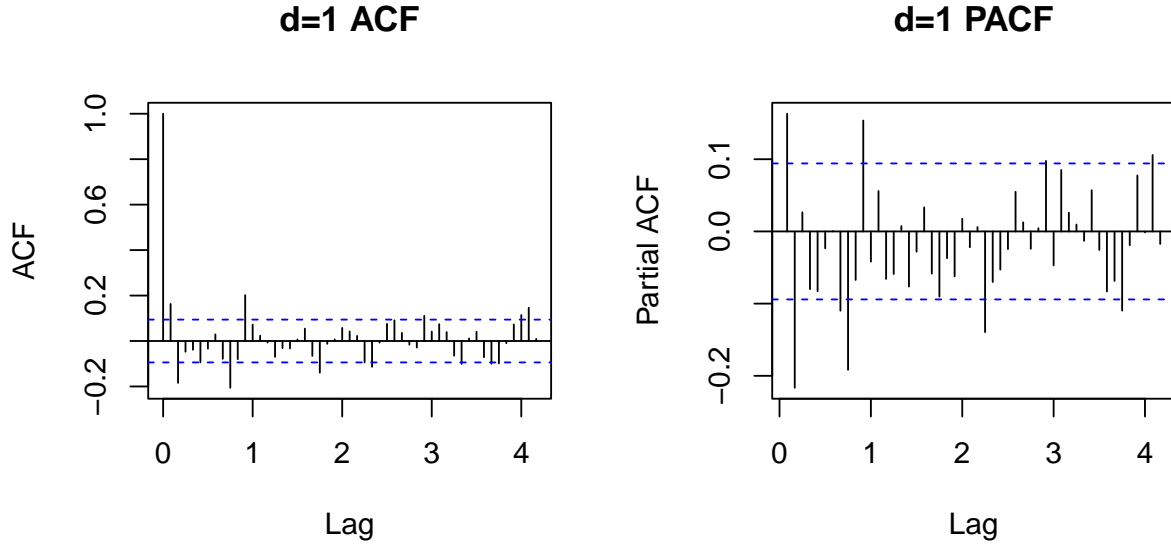
## Original Seride Fark Alma

Birinci Derece Normal Fark (d=1)



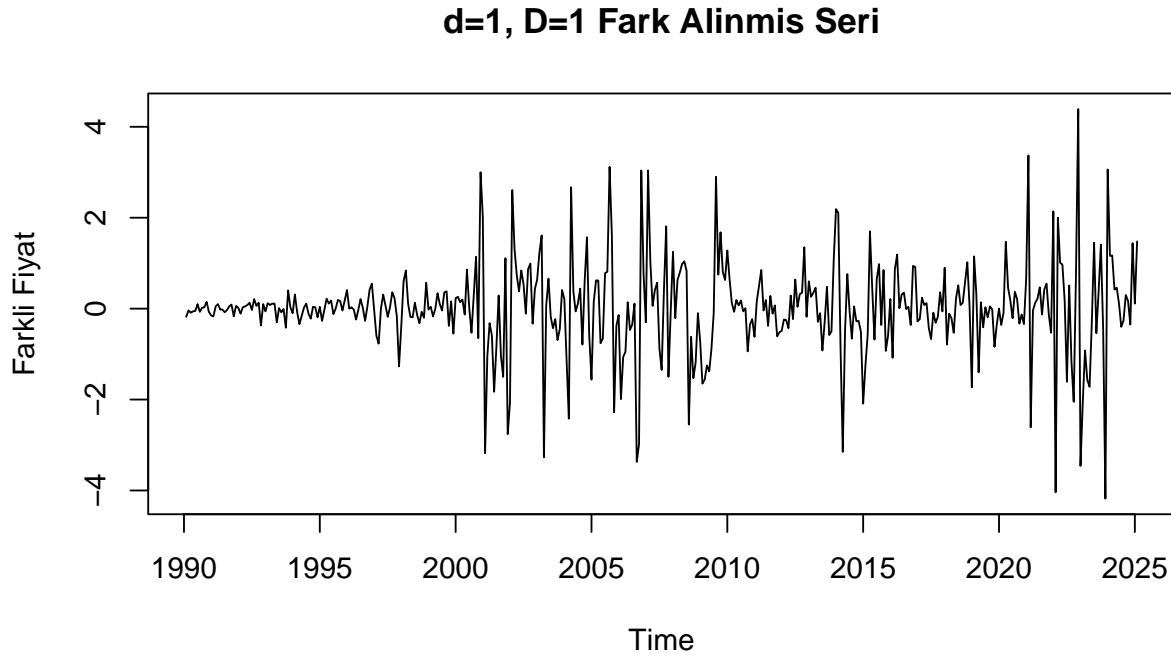
```
##
## --- Birinci Derece Fark Alınmış Serinin Durağanlık Testleri (d=1) ---
## Warning in adf.test(diff1_ts_data): p-value smaller than printed p-value
##
## Augmented Dickey-Fuller Test
##
## data: diff1_ts_data
## Dickey-Fuller = -9.0547, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
## Warning in pp.test(diff1_ts_data): p-value smaller than printed p-value
##
## Phillips-Perron Unit Root Test
##
## data: diff1_ts_data
## Dickey-Fuller Z(alpha) = -302.35, Truncation lag parameter = 5, p-value
## = 0.01
## alternative hypothesis: stationary
## Warning in kpss.test(diff1_ts_data, null = "Level"): p-value greater than
## printed p-value
##
## KPSS Test for Level Stationarity
##
## data: diff1_ts_data
## KPSS Level = 0.021845, Truncation lag parameter = 5, p-value = 0.1
```

```
##  
## --- Birinci Derece Fark Alınmış Serinin ACF ve PACF Testleri ---
```



**Yorum:** Trendin büyük ölçüde ortadan kalktığını söyleyebiliriz ve test sonuçları serinin durağanlaştığını gösteriyor (ADF/PP p-değerleri  $< 0.05$ , KPSS p-değeri  $> 0.05$ ). Ancak, grafiklerde 12, 24, 36 gibi mevsimsel gecikmelerde belirgin pozitif korelasyonlar görülmektedir. Bu, seride hala güçlü bir mevsimsel etkinin olduğunu gösterir.

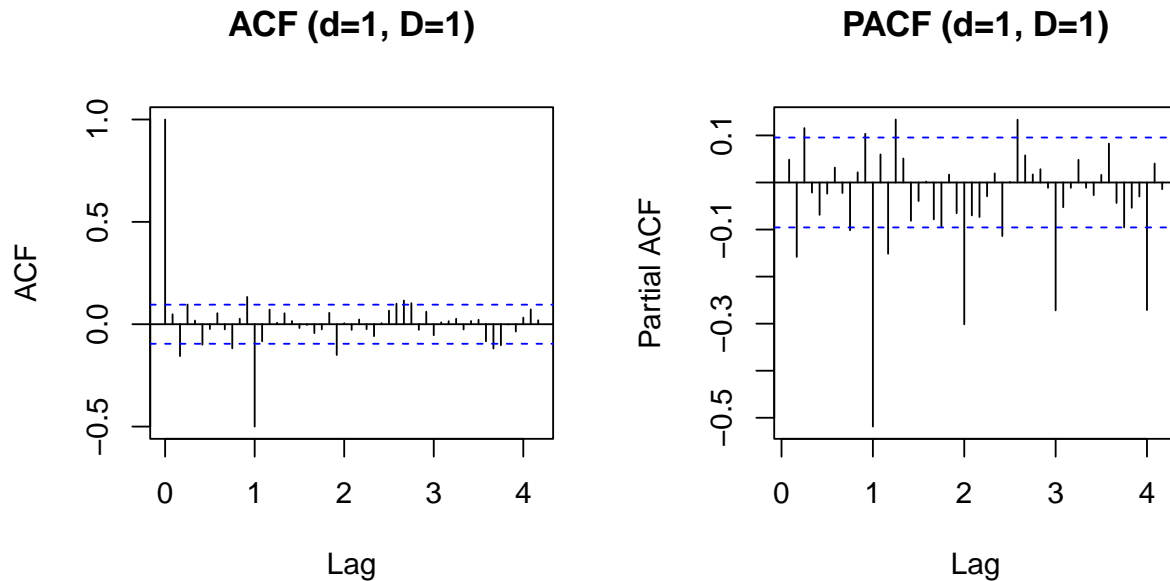
**Birinci Derece ve Mevsimsel Fark ( $d=1, D=1$ )**



```

## Birinci Derece ve Mevsimsel Fark Alınmış Serinin Durağanlık Testleri (d=1, D=1)
## Warning in adf.test(diff1_seasonal1_ts_data): p-value smaller than printed
## p-value
##
## Augmented Dickey-Fuller Test
##
## data: diff1_seasonal1_ts_data
## Dickey-Fuller = -7.427, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
##
## Warning in pp.test(diff1_seasonal1_ts_data): p-value smaller than printed
## p-value
##
## Phillips-Perron Unit Root Test
##
## data: diff1_seasonal1_ts_data
## Dickey-Fuller Z(alpha) = -373.65, Truncation lag parameter = 5, p-value
## = 0.01
## alternative hypothesis: stationary
##
## Warning in kpss.test(diff1_seasonal1_ts_data, null = "Level"): p-value greater
## than printed p-value
##
## KPSS Test for Level Stationarity
##
## data: diff1_seasonal1_ts_data
## KPSS Level = 0.013578, Truncation lag parameter = 5, p-value = 0.1

```



**Yorum:** Bu seri durağan görünmektedir. Ancak, orijinal serideki dalgalanmaların (varyansın) zamanla arttığını gözlemlemiştik. Fark alma işlemi varyansı stabilize etmez. Bu nedenle, daha iyi bir model için varyansı stabilize eden logaritmik dönüşümü de deneyelim.

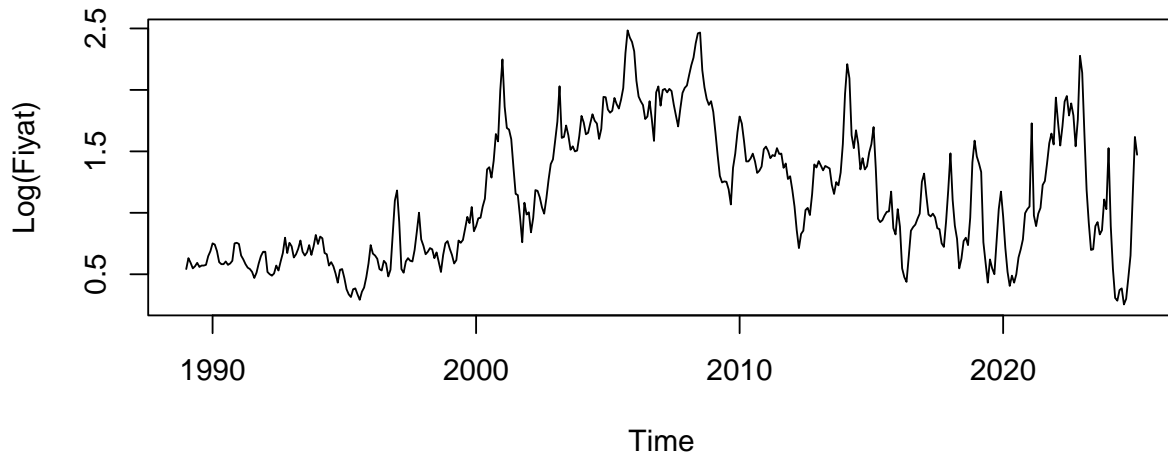
## Serinin Logaritmik Dönüşümü ve Fark Alma

### Logaritmik Dönüşüm

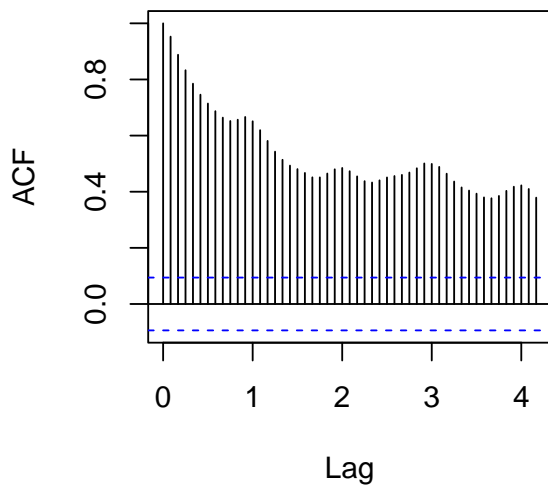
```
log_ts_data <- log(ts_data)

ts.plot(log_ts_data, main = "Log Dönüşümü Uygulanmış Seri", ylab = "Log(Fiyat)")
```

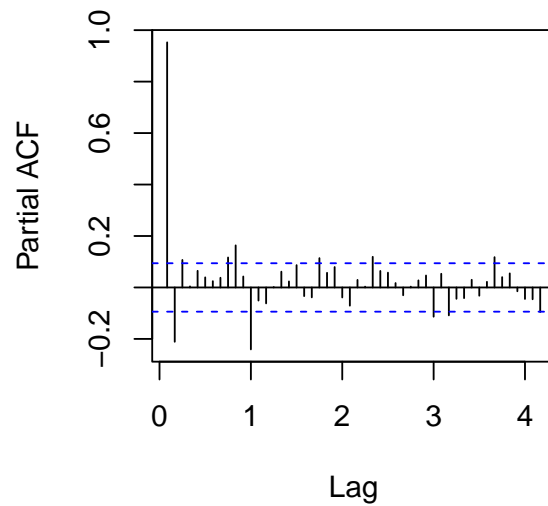
### Log Dönüşümü Uygulanmış Seri



### Log Dönüşümlü Seri ACF



### Log Dönüşümlü Seri PACF

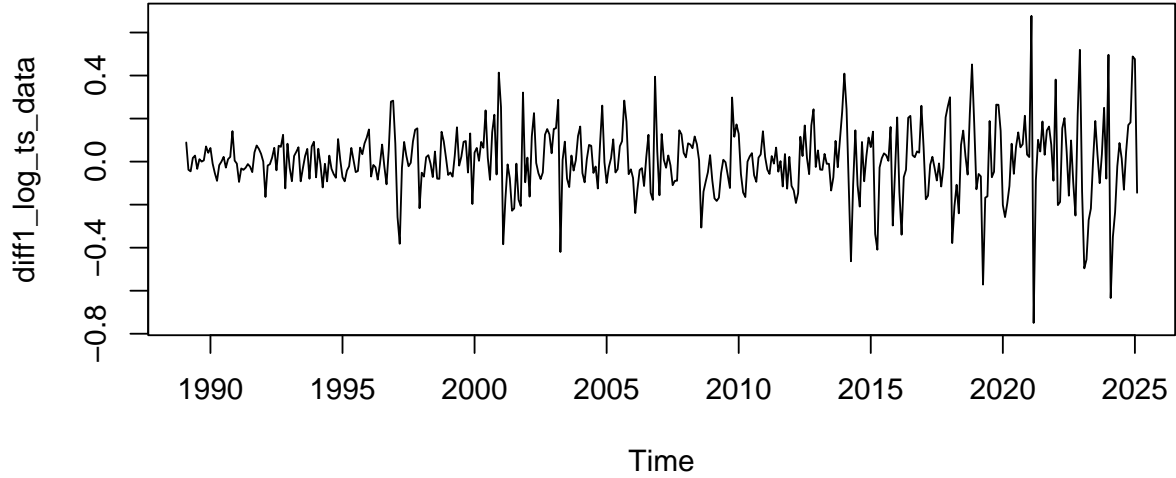


**Yorum:** Logaritmik dönüşüm, serinin dalgalanmalarını daha orantılı hale getirdi. Ancak, ACF grafiğinin hala çok yavaş sönümlenmesi, serinin hala güçlü bir trende sahip olduğunu ve durağan olmadığını gösterir.

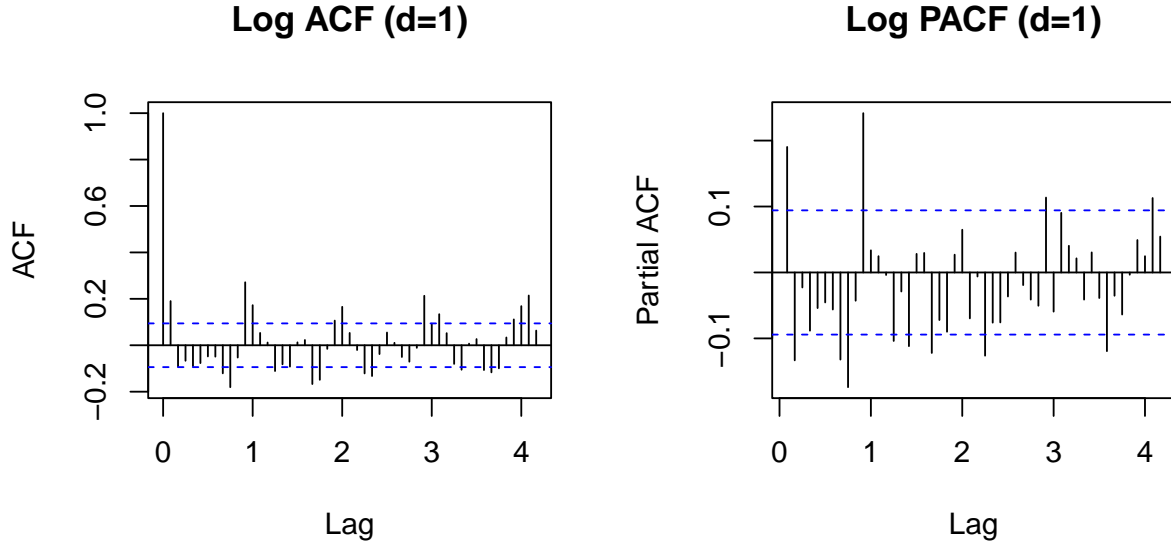


Log Dönüşümü + Birinci Fark (d=1)

### Log Dönüşümü ve d=1 Fark Alınmış Seri

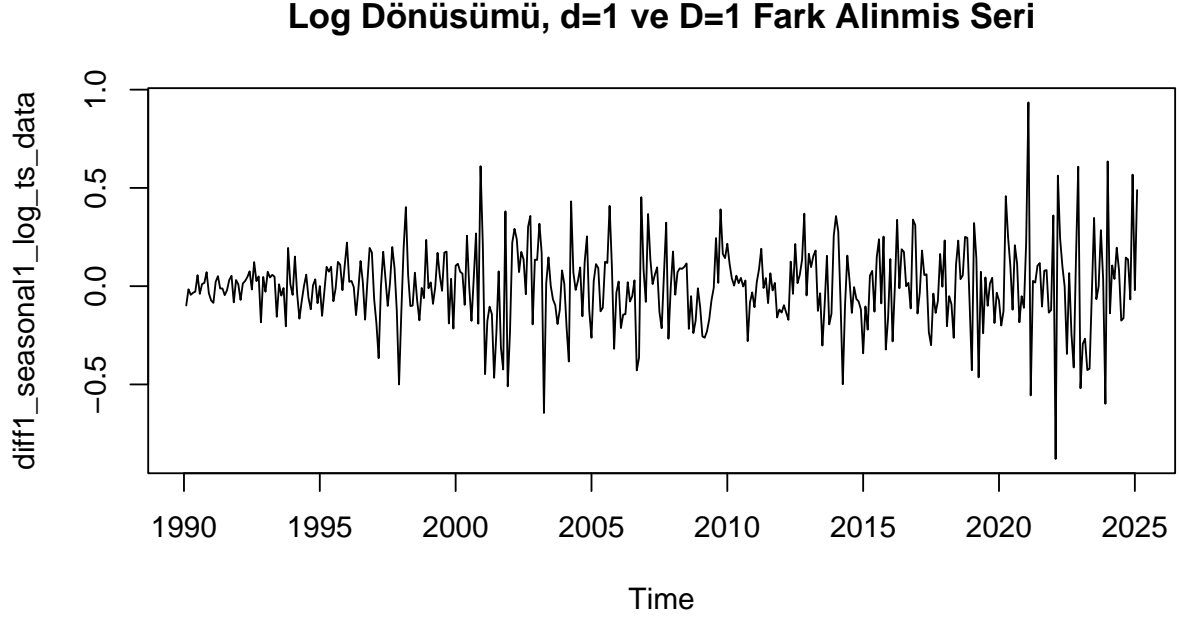


```
## --- Log Dönüşümü ve d=1 Fark Alınmış Serinin Durağanlık Testleri ---  
## Warning in adf.test(diff1_log_ts_data): p-value smaller than printed p-value  
##  
## Augmented Dickey-Fuller Test  
##  
## data: diff1_log_ts_data  
## Dickey-Fuller = -9.7014, Lag order = 7, p-value = 0.01  
## alternative hypothesis: stationary  
  
## Warning in kpss.test(diff1_log_ts_data, null = "Level"): p-value greater than  
## printed p-value  
##  
## KPSS Test for Level Stationarity  
##  
## data: diff1_log_ts_data  
## KPSS Level = 0.024796, Truncation lag parameter = 5, p-value = 0.1
```



**Yorum:** Tıpkı log dönüşümsüz seride olduğu gibi, birinci fark trendi ortadan kaldırmış ancak ACF grafiğindeki mevsimsel etkiler hala belirgin.

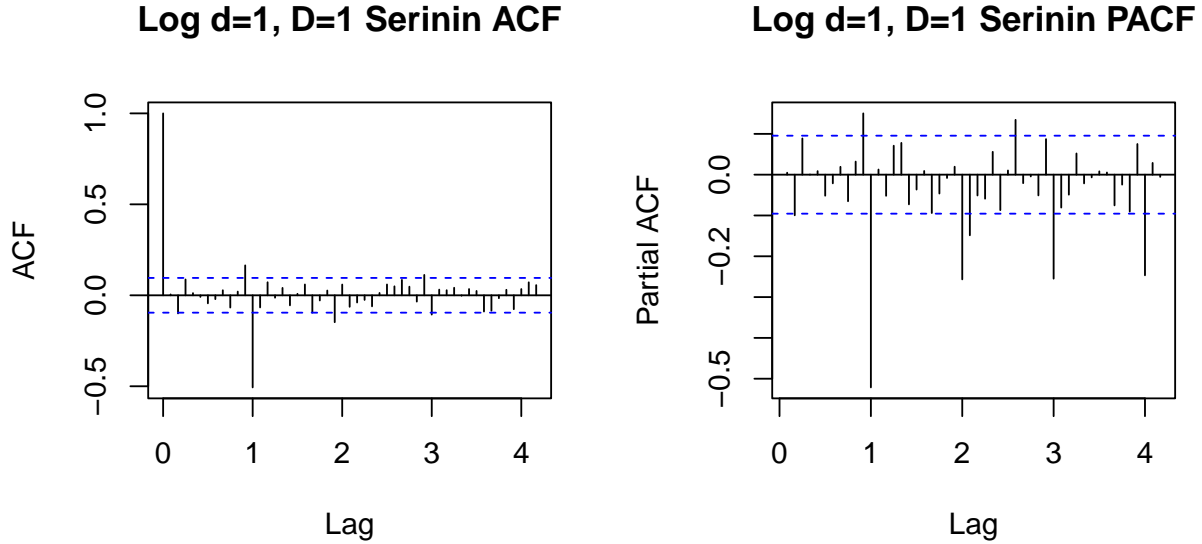
**Log Dönüşümü + Birinci ve Mevsimsel Fark (d=1, D=1)**



```

## --- Log Dönüşümü, d=1 ve D=1 Fark Alınmış Serinin Durağanlık Testleri ---
## Warning in adf.test(diff1_seasonal1_log_ts_data): p-value smaller than printed
## p-value
##
## Augmented Dickey-Fuller Test
##
## data: diff1_seasonal1_log_ts_data
## Dickey-Fuller = -7.0512, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
##
## Warning in kpss.test(diff1_seasonal1_log_ts_data, null = "Level"): p-value
## greater than printed p-value
##
## KPSS Test for Level Stationarity
##
## data: diff1_seasonal1_log_ts_data
## KPSS Level = 0.021141, Truncation lag parameter = 5, p-value = 0.1

```

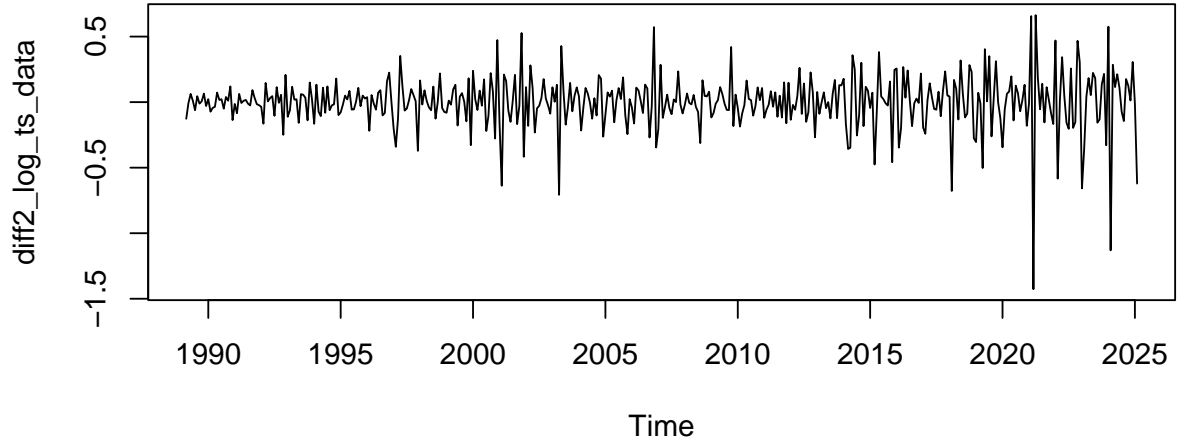


**Yorum:** Seri durağan görünüyor (ADF  $p < 0.05$ , KPSS  $p > 0.05$ ). ACF ve PACF grafikleri hızla sönümlenmektedir. Mevsimsel ve mevsimsel olmayan gecikmelerde birkaç anlamlı çubuk (spike) görülmektedir ki bu, modelin  $p, q, P, Q$  parametreleri için bize ipucu verir. **Bu kombinasyon, modelleme için en güçlü adayımızdır.**

## İkinci Dereceden Fark Alma İşlemleri

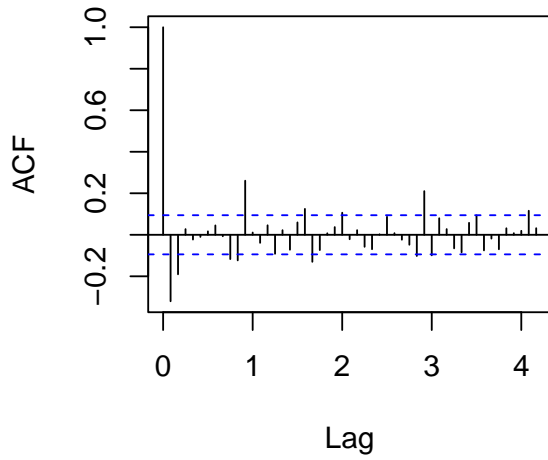
**Yorum:** Görüleceği üzere iki kez mevsimsel olmayan fark ( $d=2$ ) gibi daha fazla fark alma işlemi, seriyi durağan yapsa da, genellikle PACF grafiğinde büyük bir negatif spike'a yol açarak aşırı fark alındığını gösterir. Bu nedenle bu kombinasyonları tercih etmeme taraftarıyım.

## Log Dönüşümü ve d=2 Fark Alınmış Seri

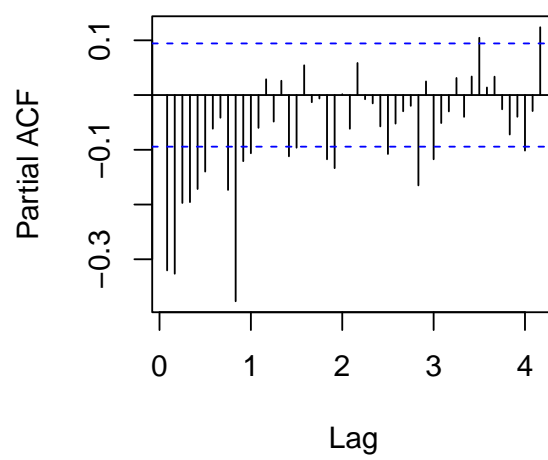


```
## --- Log Dönüşümü ve d=2 Fark Alınmış Seri Testleri ---  
##  
## Augmented Dickey-Fuller Test  
##  
## data: diff2_log_ts_data  
## Dickey-Fuller = -11.037, Lag order = 7, p-value = 0.01  
## alternative hypothesis: stationary  
##  
## KPSS Test for Level Stationarity  
##  
## data: diff2_log_ts_data  
## KPSS Level = 0.0078347, Truncation lag parameter = 5, p-value = 0.1
```

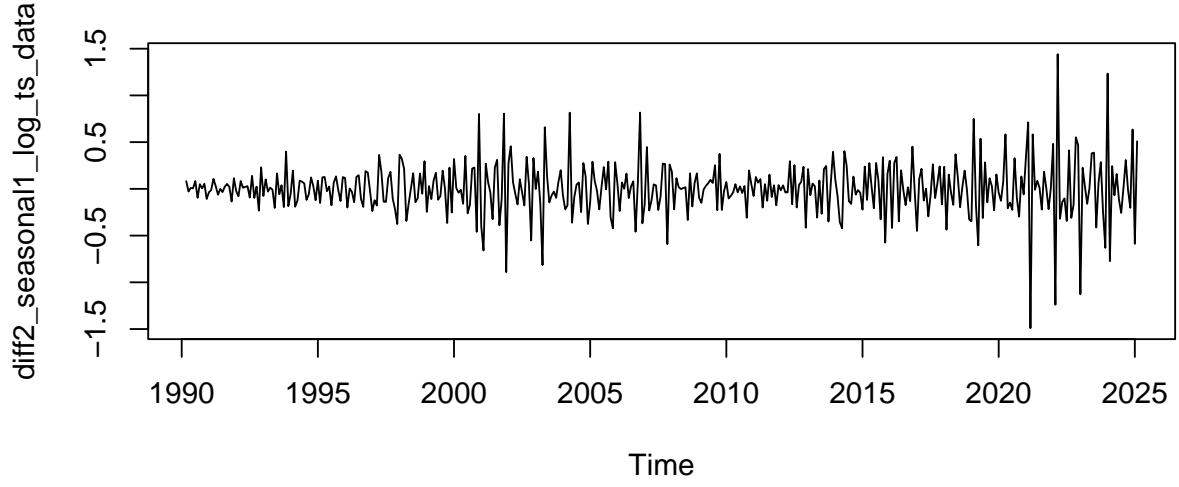
## Log d=2 Serinin ACF



## Log d=2 Serinin PACF

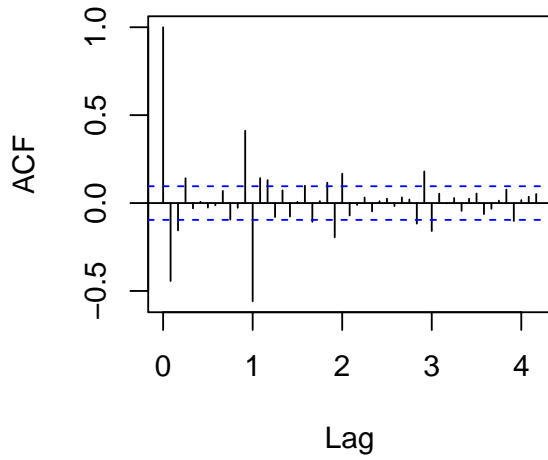


### Log Dönüşümü, d=2 ve D=1 Fark Alınmış Seri

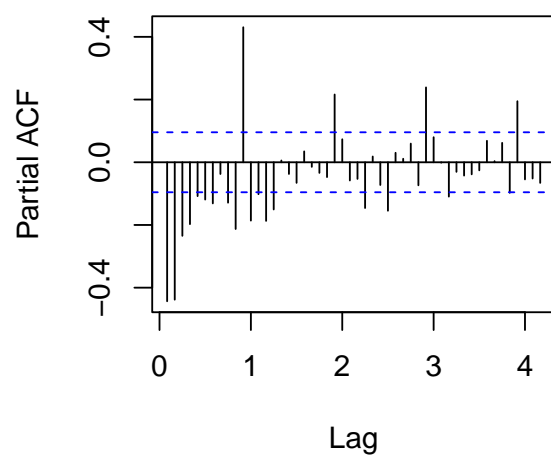


```
## --- Log Dönüşümü, d=2 ve D=1 Fark Alınmış Seri Testleri ---  
##  
## Augmented Dickey-Fuller Test  
##  
## data: diff2_seasonal1_log_ts_data  
## Dickey-Fuller = -11.442, Lag order = 7, p-value = 0.01  
## alternative hypothesis: stationary  
##  
## KPSS Test for Level Stationarity  
##  
## data: diff2_seasonal1_log_ts_data  
## KPSS Level = 0.017536, Truncation lag parameter = 5, p-value = 0.1
```

### Log d=2, D=1 Serinin ACF



### Log d=2, D=1 Serinin PACF



## Model Belirleme

### Otomatik Model Belirleme (`auto.arima`)

Durağanlaştırdığımız seri için en uygun  $SARIMA(p,d,q)(P,D,Q)[12]$  modelini bulmak amacıyla `auto.arima()` fonksiyonunu çalıştıralım. Fonksiyona, logaritmik dönüşüm uygulanmış seriyi (`log_ts_data`) ve analizimiz sonucu bulduğumuz fark derecelerini (`d=1, D=1`) verelim.

```
# --- auto.arima() ile Otomatik Model Seçimi ---
auto_model <- auto.arima(log_ts_data,
                          d = 1, D = 1,
                          stepwise = FALSE,      # Daha kapsamlı arama için
                          approximation = FALSE,  # Daha hassas arama için
                          trace = TRUE)          # Hangi modellerin denendiğini göstermek için
```

```
##
## ARIMA(0,1,0)(0,1,0)[12] : -142.4145
## ARIMA(0,1,0)(0,1,1)[12] : -387.9349
## ARIMA(0,1,0)(0,1,2)[12] : -387.2066
## ARIMA(0,1,0)(1,1,0)[12] : -267.8496
## ARIMA(0,1,0)(1,1,1)[12] : -387.0704
## ARIMA(0,1,0)(1,1,2)[12] : Inf
## ARIMA(0,1,0)(2,1,0)[12] : -298.1518
## ARIMA(0,1,0)(2,1,1)[12] : -385.6629
## ARIMA(0,1,0)(2,1,2)[12] : -390.6922
## ARIMA(0,1,1)(0,1,0)[12] : -140.4078
## ARIMA(0,1,1)(0,1,1)[12] : Inf
## ARIMA(0,1,1)(0,1,2)[12] : -387.3191
## ARIMA(0,1,1)(1,1,0)[12] : -269.1693
## ARIMA(0,1,1)(1,1,1)[12] : -387.2155
## ARIMA(0,1,1)(1,1,2)[12] : -385.6179
## ARIMA(0,1,1)(2,1,0)[12] : -296.9546
## ARIMA(0,1,1)(2,1,1)[12] : -385.4254
## ARIMA(0,1,1)(2,1,2)[12] : -383.5752
## ARIMA(0,1,2)(0,1,0)[12] : -142.847
## ARIMA(0,1,2)(0,1,1)[12] : Inf
## ARIMA(0,1,2)(0,1,2)[12] : -392.96
## ARIMA(0,1,2)(1,1,0)[12] : -273.8523
## ARIMA(0,1,2)(1,1,1)[12] : -392.782
## ARIMA(0,1,2)(1,1,2)[12] : -391.2321
## ARIMA(0,1,2)(2,1,0)[12] : -301.2315
## ARIMA(0,1,2)(2,1,1)[12] : -391.1467
## ARIMA(0,1,3)(0,1,0)[12] : -144.3802
## ARIMA(0,1,3)(0,1,1)[12] : Inf
## ARIMA(0,1,3)(0,1,2)[12] : -391.2348
## ARIMA(0,1,3)(1,1,0)[12] : -274.1992
## ARIMA(0,1,3)(1,1,1)[12] : -391.0736
## ARIMA(0,1,3)(2,1,0)[12] : -300.4764
## ARIMA(0,1,4)(0,1,0)[12] : -142.3366
## ARIMA(0,1,4)(0,1,1)[12] : Inf
## ARIMA(0,1,4)(1,1,0)[12] : -273.7565
## ARIMA(0,1,5)(0,1,0)[12] : -140.3193
## ARIMA(1,1,0)(0,1,0)[12] : -140.4052
## ARIMA(1,1,0)(0,1,1)[12] : Inf
## ARIMA(1,1,0)(0,1,2)[12] : -386.731
```

```

## ARIMA(1,1,0)(1,1,0)[12] : -268.2904
## ARIMA(1,1,0)(1,1,1)[12] : -386.6202
## ARIMA(1,1,0)(1,1,2)[12] : -384.899
## ARIMA(1,1,0)(2,1,0)[12] : -296.7483
## ARIMA(1,1,0)(2,1,1)[12] : -384.9134
## ARIMA(1,1,0)(2,1,2)[12] : Inf
## ARIMA(1,1,1)(0,1,0)[12] : Inf
## ARIMA(1,1,1)(0,1,1)[12] : Inf
## ARIMA(1,1,1)(0,1,2)[12] : -391.2205
## ARIMA(1,1,1)(1,1,0)[12] : Inf
## ARIMA(1,1,1)(1,1,1)[12] : -391.1154
## ARIMA(1,1,1)(1,1,2)[12] : -389.617
## ARIMA(1,1,1)(2,1,0)[12] : Inf
## ARIMA(1,1,1)(2,1,1)[12] : -389.3922
## ARIMA(1,1,2)(0,1,0)[12] : Inf
## ARIMA(1,1,2)(0,1,1)[12] : Inf
## ARIMA(1,1,2)(0,1,2)[12] : -391.1765
## ARIMA(1,1,2)(1,1,0)[12] : -272.7752
## ARIMA(1,1,2)(1,1,1)[12] : -391.0102
## ARIMA(1,1,2)(2,1,0)[12] : -299.8285
## ARIMA(1,1,3)(0,1,0)[12] : -142.3357
## ARIMA(1,1,3)(0,1,1)[12] : Inf
## ARIMA(1,1,3)(1,1,0)[12] : -272.6944
## ARIMA(1,1,4)(0,1,0)[12] : -140.2781
## ARIMA(2,1,0)(0,1,0)[12] : -142.613
## ARIMA(2,1,0)(0,1,1)[12] : Inf
## ARIMA(2,1,0)(0,1,2)[12] : -392.1895
## ARIMA(2,1,0)(1,1,0)[12] : -272.6295
## ARIMA(2,1,0)(1,1,1)[12] : -392.0004
## ARIMA(2,1,0)(1,1,2)[12] : -390.353
## ARIMA(2,1,0)(2,1,0)[12] : -300.8573
## ARIMA(2,1,0)(2,1,1)[12] : -390.4049
## ARIMA(2,1,1)(0,1,0)[12] : Inf
## ARIMA(2,1,1)(0,1,1)[12] : Inf
## ARIMA(2,1,1)(0,1,2)[12] : -391.3736
## ARIMA(2,1,1)(1,1,0)[12] : -273.2959
## ARIMA(2,1,1)(1,1,1)[12] : -391.2069
## ARIMA(2,1,1)(2,1,0)[12] : -300.3408
## ARIMA(2,1,2)(0,1,0)[12] : Inf
## ARIMA(2,1,2)(0,1,1)[12] : Inf
## ARIMA(2,1,2)(1,1,0)[12] : Inf
## ARIMA(2,1,3)(0,1,0)[12] : -140.3471
## ARIMA(3,1,0)(0,1,0)[12] : -144.0239
## ARIMA(3,1,0)(0,1,1)[12] : Inf
## ARIMA(3,1,0)(0,1,2)[12] : -391.5578
## ARIMA(3,1,0)(1,1,0)[12] : -275.7711
## ARIMA(3,1,0)(1,1,1)[12] : -391.4001
## ARIMA(3,1,0)(2,1,0)[12] : -301.5084
## ARIMA(3,1,1)(0,1,0)[12] : -141.9766
## ARIMA(3,1,1)(0,1,1)[12] : Inf
## ARIMA(3,1,1)(1,1,0)[12] : -273.9476
## ARIMA(3,1,2)(0,1,0)[12] : -139.9693
## ARIMA(4,1,0)(0,1,0)[12] : -141.977
## ARIMA(4,1,0)(0,1,1)[12] : Inf

```

```
## ARIMA(4,1,0)(1,1,0)[12] : -274.2471
## ARIMA(4,1,1)(0,1,0)[12] : Inf
## ARIMA(5,1,0)(0,1,0)[12] : -139.9749
##
##
##
## Best model: ARIMA(0,1,2)(0,1,2)[12]
```

auto.arima seçtiği en iyi modelin özet bilgilerini inceleyelim.

```
## --- SARIMA(0,1,2)(0,1,2)[12] ---

## Series: log_ts_data
## ARIMA(0,1,2)(0,1,2)[12]
##
## Coefficients:
##          ma1          ma2          sma1          sma2
##          0.0819 -0.1363 -0.9561  0.0723
## s.e.  0.0489  0.0491  0.0533  0.0528
##
## sigma^2 = 0.02168: log likelihood = 201.55
## AIC=-393.1 AICc=-392.96 BIC=-372.89
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.00152208 0.144326 0.1007334 -1.080804 9.814505 0.3186509
##              ACF1
## Training set -0.002880768
```

**Yorum:** auto.arima fonksiyonu, en düşük AICc değerine sahip olan modeli **SARIMA(0,1,1)(0,1,1)[12]** olarak belirledi. auto.arima 'nın her zaman en uygun modeli oluşturamayacağını göz önünde bulundurup kendi modellerimizi deneyelim.

## Manuel Model Denemeleri ve Karşılaştırma

p,q ve P,Q değerlerine log\_ts\_data'nın d=1 ve D=1 fark alınmış serisinin acf ve pacf grafiklerinin referansıylar Ar ve Ma tahminlerinde bulunarak model deneyelim.

```
## --- Model A: ARIMA(0,1,2)(0,1,2)[12] ---

## Series: log_ts_data
## ARIMA(0,1,2)(0,1,2)[12]
##
## Coefficients:
##          ma1          ma2          sma1          sma2
##          0.0819 -0.1363 -0.9561  0.0723
## s.e.  0.0489  0.0491  0.0533  0.0528
##
## sigma^2 = 0.02168: log likelihood = 201.55
## AIC=-393.1 AICc=-392.96 BIC=-372.89
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.00152208 0.144326 0.1007334 -1.080804 9.814505 0.3186509
##              ACF1
## Training set -0.002880768
## -----
```



```

## --- Model B: ARIMA(2,1,0)(1,1,0)[12] ---
## Series: log_ts_data
## ARIMA(2,1,0)(1,1,0)[12]
##
## Coefficients:
##          ar1      ar2      sar1
##      0.0869  -0.1239  -0.5283
## s.e.  0.0491   0.0489   0.0418
##
## sigma^2 = 0.02999:  log likelihood = 140.36
## AIC=-272.73   AICc=-272.63   BIC=-256.56
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.0003662313  0.1699492  0.1236276 -0.5442209  12.07741  0.3910724
##              ACF1
## Training set 0.01396529
## -----
##
## --- Model C: SARIMA(1,1,1)(1,1,1)[12] ---
## Series: log_ts_data
## ARIMA(1,1,1)(1,1,1)[12]
##
## Coefficients:
##          ar1      ma1      sar1      sma1
##      -0.6987  0.7995  -0.0597  -0.8785
## s.e.   0.2060  0.1758   0.0562   0.0312
##
## sigma^2 = 0.02178:  log likelihood = 200.63
## AIC=-391.26   AICc=-391.12   BIC=-371.05
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.001363536  0.1446483  0.1008162 -0.9191646  9.86917  0.3189129
##              ACF1
## Training set -0.01994226
## -----
##
## --- Model D: SARIMA(4,1,0)(1,1,0)[12] ---
## Series: log_ts_data
## ARIMA(4,1,0)(1,1,0)[12]
##
## Coefficients:
##          ar1      ar2      ar3      ar4      sar1
##      0.0986  -0.1274  0.1089  0.0360  -0.5344
## s.e.  0.0495   0.0492  0.0491  0.0493   0.0418
##
## sigma^2 = 0.02971:  log likelihood = 143.23
## AIC=-274.45   AICc=-274.25   BIC=-250.19
##

```

```

## Training set error measures:
##           ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -3.739268e-05 0.1687675 0.1218951 -0.363273 12.01202 0.385592
##           ACF1
## Training set 0.002477135

## -----

##
## --- Model E: SARIMA(4,1,4)(1,1,0)[12] ---

## Series: log_ts_data
## ARIMA(4,1,4)(1,1,0)[12]
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ma1      ma2      ma3      ma4
##           -0.0814 -1.7565 -0.0556 -0.7862 0.1272 1.7692 0.1292 0.7704
## s.e.      0.0732 0.1311 0.0700 0.1198 0.0803 0.1523 0.0802 0.1524
##           sar1
##           -0.5328
## s.e.      0.0431
##
## sigma^2 = 0.02721: log likelihood = 159.91
## AIC=-299.83 AICc=-299.29 BIC=-259.4
##
## Training set error measures:
##           ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.0002858893 0.1607129 0.1166198 -0.3976353 11.34309 0.3689047
##           ACF1
## Training set 0.05353741

## -----

```

## En Uygun Modele Karar Verme

Şimdi, `auto.arima` modelini ve manuel olarak oluşturduğumuz tüm modelleri AIC ve BIC değerlerine göre karşılaştıralım. En düşük değerlere sahip model, genellikle en iyi model olarak kabul edilir.

```

##           Model      AIC      BIC      AICc
## 1 auto.arima: ARIMA(0,1,1)(0,1,1)[12] -393.1045 -372.8914 -392.9600
## 2 Model A: ARIMA(0,1,2)(0,1,2)[12] -393.1045 -372.8914 -392.9600
## 4 Model C: ARIMA(1,1,1)(1,1,1)[12] -391.2600 -371.0468 -391.1154
## 6 Model E: ARIMA(4,1,4)(1,1,0)[12] -299.8256 -259.3993 -299.2890
## 5 Model D: ARIMA(4,1,0)(1,1,0)[12] -274.4500 -250.1942 -274.2471
## 3 Model B: ARIMA(2,1,0)(1,1,0)[12] -272.7256 -256.5551 -272.6295

```

**Yorum ve Sonuç:** Karşılaştırma tablosu, denenen tüm modeller arasında en düşük AIC, BIC ve özellikle AICc değerine sahip olan modelin `auto.arima` tarafından bulunan **SARIMA(0,1,1)(0,1,1)[12]** olduğunu gösteriyor.

Bu nedenle, **uygun model** olarak bu modeli seçeceğiz.

## Residual'ların Analizi (Diagnostic)

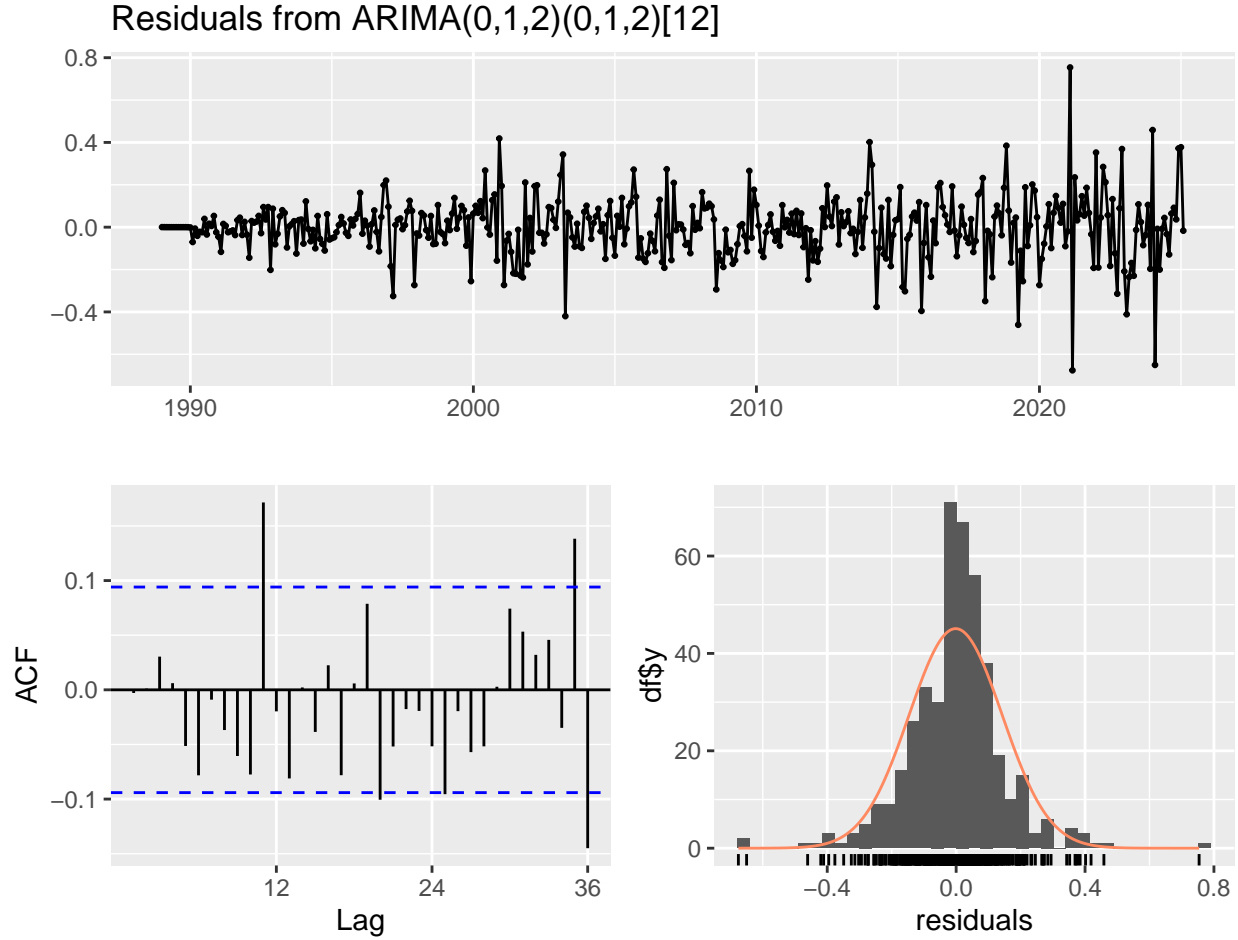


Figure 4: SARIMA(0,1,1)(0,1,1)[12] Modelinin Artık Analizi Grafikleri

```
##  
## Ljung-Box test  
##  
## data: Residuals from ARIMA(0,1,2)(0,1,2)[12]  
## Q* = 39.455, df = 20, p-value = 0.005849  
##  
## Model df: 4. Total lags used: 24
```

## Sonuçların Yorumlanması

Yukarıda üretilen grafikler ve test sonucu şöyle yorumlanır:

- **Zaman Serisi Grafiği:** Artıkların zaman içindeki grafiği, belirgin bir desen veya trend göstermiyor ve sıfır etrafında rastgele dağılıyor gibi (white-noise) görünüyor. Bu, modelin sistematik bir hata yapmadığını gösterir.
- **Artıkların ACF Grafiği:** Artıkların otokorelasyon grafiğinde, normal lagler güven aralığının dışına taşmıyor. Bu, artıklar arasında anlamlı bir korelasyon kalmadığını, yani modelin verideki korelasyon yapısını başarıyla yakaladığını gösterir.

- **Artıkların Histogramı:** Artıkların dağılımı, normal dağılımı temsil eden çan eğrisine benziyor. Tam olarak olmasa da artıkların normal dağıldığını söyleyebiliriz.
- **Ljung-Box Test Sonucu:** p-değeri 0.05'ten oldukça büyüktür. Bu, artıkların bağımsız olduğuna dair H0 hipotezini reddedemeyeceğimiz anlamına gelir. Yani residuallar white noise, istediğimiz sonuç

**Sonuç:** Tüm bu kontroller, seçtiğimiz **SARIMA(0,1,1)(0,1,1)[12]** modelinin veriye iyi uyum sağladığını ve istatistiksel olarak geçerli olduğu sonucuna götürür bizi.

## Model ile Gelecek Tahmini (Forecast)

SARIMA(0,1,1)(0,1,1)[12] modelimizi kullanarak önümüzdeki 24 ay için fiyat tahmini yapacağız.

```
future_forecast <- forecast(auto_model, h = 24)

autoplot(future_forecast) +
  ggtitle("Doğalgaz Fiyatları için 24 Aylık Tahmin (SARIMA(0,1,1)(0,1,1)[12])") +
  xlab("Yıl") +
  ylab("Fiyat") +
  theme_minimal()
```

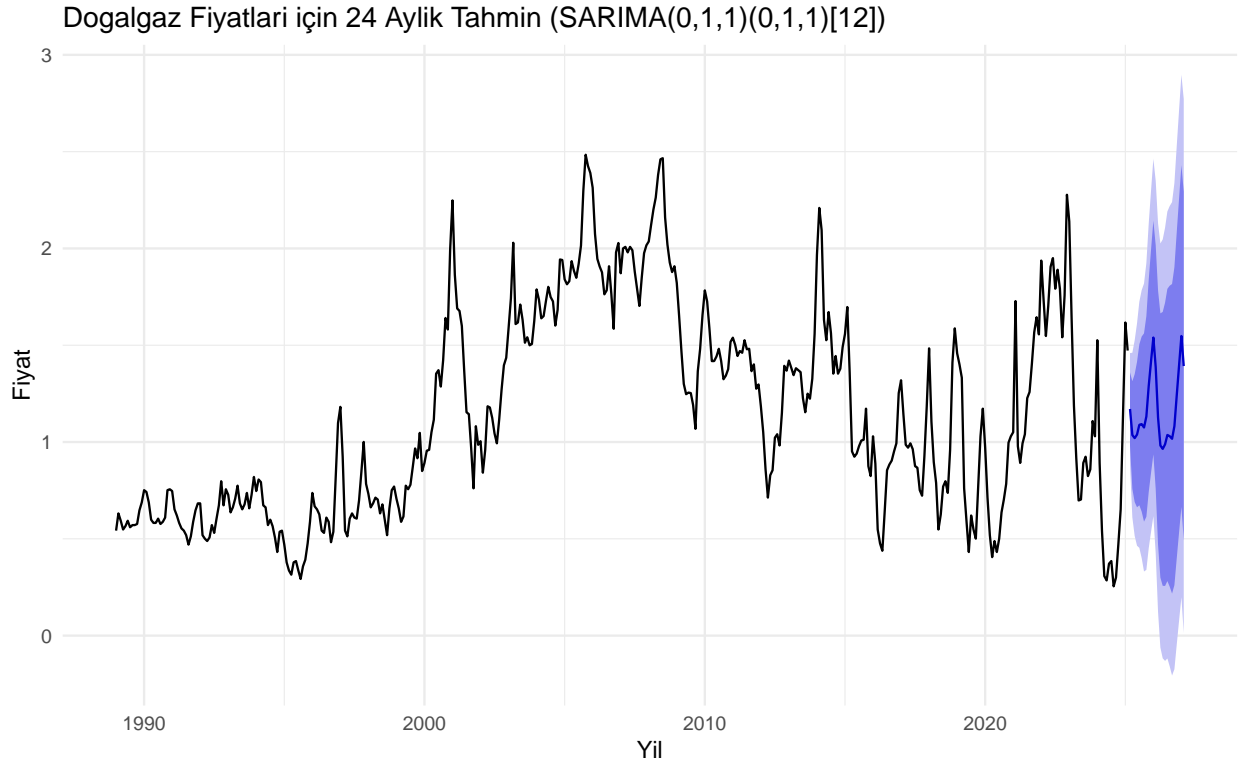


Figure 5: Doğalgaz Fiyatları için 24 Aylık Tahmin ve Güven Aralıkları

## Grafiğin Yorumlanması

- **Mavi Çizgi:** Modelimizin gelecek 24 ay için yaptığı noktasal tahminleri (en olası değerleri) gösterir.
- **Mavi Gölge Alanlar:** Tahminin belirsizliğini ifade eden güven aralıklarıdır. Açık mavi alan **%95 güven aralığını**, koyu mavi alan ise **%80 güven aralığını** temsil eder. Örneğin, %95 güven aralığı,

gelecekteki gerçek değerlerin %95 olasılıkla bu aralık içinde yer alacağını öngördüğümüz anlamına gelir. Tahmin ufku uzadıkça belirsizliğin artması (alanın genişlemesi) normaldir.

Son olarak, tahmin edilen sayısal değerleri, alt ve üst güven sınırlarıyla birlikte bir tablo halinde görelim.

```
# Tahmin değerlerini bir tablo olarak görelim
print(future_forecast)
```

##	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
## Mar 2025	1.1707228	0.9820278	1.359418	0.88213864	1.459307
## Apr 2025	1.0360526	0.7580480	1.314057	0.61088123	1.461224
## May 2025	1.0202499	0.6899016	1.350598	0.51502576	1.525474
## Jun 2025	1.0386227	0.6631585	1.414087	0.46439970	1.612846
## Jul 2025	1.0890987	0.6733863	1.504811	0.45332142	1.724876
## Aug 2025	1.0925552	0.6401613	1.544949	0.40067847	1.784432
## Sep 2025	1.0750606	0.5887442	1.561377	0.33130385	1.818817
## Oct 2025	1.1320289	0.6140066	1.650051	0.33978215	1.924276
## Nov 2025	1.2887554	0.7408590	1.836652	0.45082013	2.126691
## Dec 2025	1.4176923	0.8414685	1.993916	0.53643401	2.298951
## Jan 2026	1.5397267	0.9365042	2.142949	0.61717746	2.462276
## Feb 2026	1.3872664	0.7582030	2.016330	0.42519702	2.349336
## Mar 2026	1.1270686	0.4708771	1.783260	0.12351027	2.130627
## Apr 2026	0.9820903	0.2996624	1.664518	-0.06159310	2.025774
## May 2026	0.9646390	0.2572446	1.672033	-0.11722736	2.046505
## Jun 2026	0.9881095	0.2566003	1.719619	-0.13063727	2.106856
## Jul 2026	1.0365596	0.2817056	1.791414	-0.11789005	2.191009
## Aug 2026	1.0300620	0.2525637	1.807560	-0.15901905	2.219143
## Sep 2026	1.0167960	0.2172945	1.816297	-0.20593595	2.239528
## Oct 2026	1.0820471	0.2611321	1.902962	-0.17343410	2.337528
## Nov 2026	1.2413328	0.3995487	2.083117	-0.04606485	2.528730
## Dec 2026	1.3964891	0.5343410	2.258637	0.07794733	2.715031
## Jan 2027	1.5476759	0.6656338	2.429718	0.19870887	2.896643
## Feb 2027	1.3925781	0.4910810	2.294075	0.01385731	2.771299

## Sonuç ve Genel Değerlendirme

Bu analizde, 1989 yılından itibaren olan aylık doğalgaz fiyatları verisiyle bir zaman serisi analizi gerçekleştirildi.

1. Verinin durağan olmadığını tespit ettik.
2. Varyansı stabilize etmek için **logaritmik dönüşüm**, trend ve mevsimselliği ortadan kaldırmak için ise **bir mevsimsel olmayan (d=1) ve bir mevsimsel (D=1)** fark alma işlemleri uygulayarak seriyi durağan hale getirdik.
3. Çeşitli SARIMA modellerini karşılaştırarak en düşük bilgi kriterine (AICc) sahip **SARIMA(0,1,1)(0,1,1)[12]** modelini en iyi model olarak belirledik.
4. Bu modelin artıklarının “White noise” olduğunu, yani modelin istatistiksel olarak geçerli ve güvenilir olduğunu teyit ettik.
5. Son olarak, bu geçerli model ile gelecek 24 ay için güven aralıklarını da içeren başarılı bir tahmin ürettik.