



IN14/2021 SARA SAPUNDŽIJA

TIME SERIES: ONLINE RETAIL



SADRŽAJ

SKUP PODATAKA

PROCESIRANJE PODATAKA

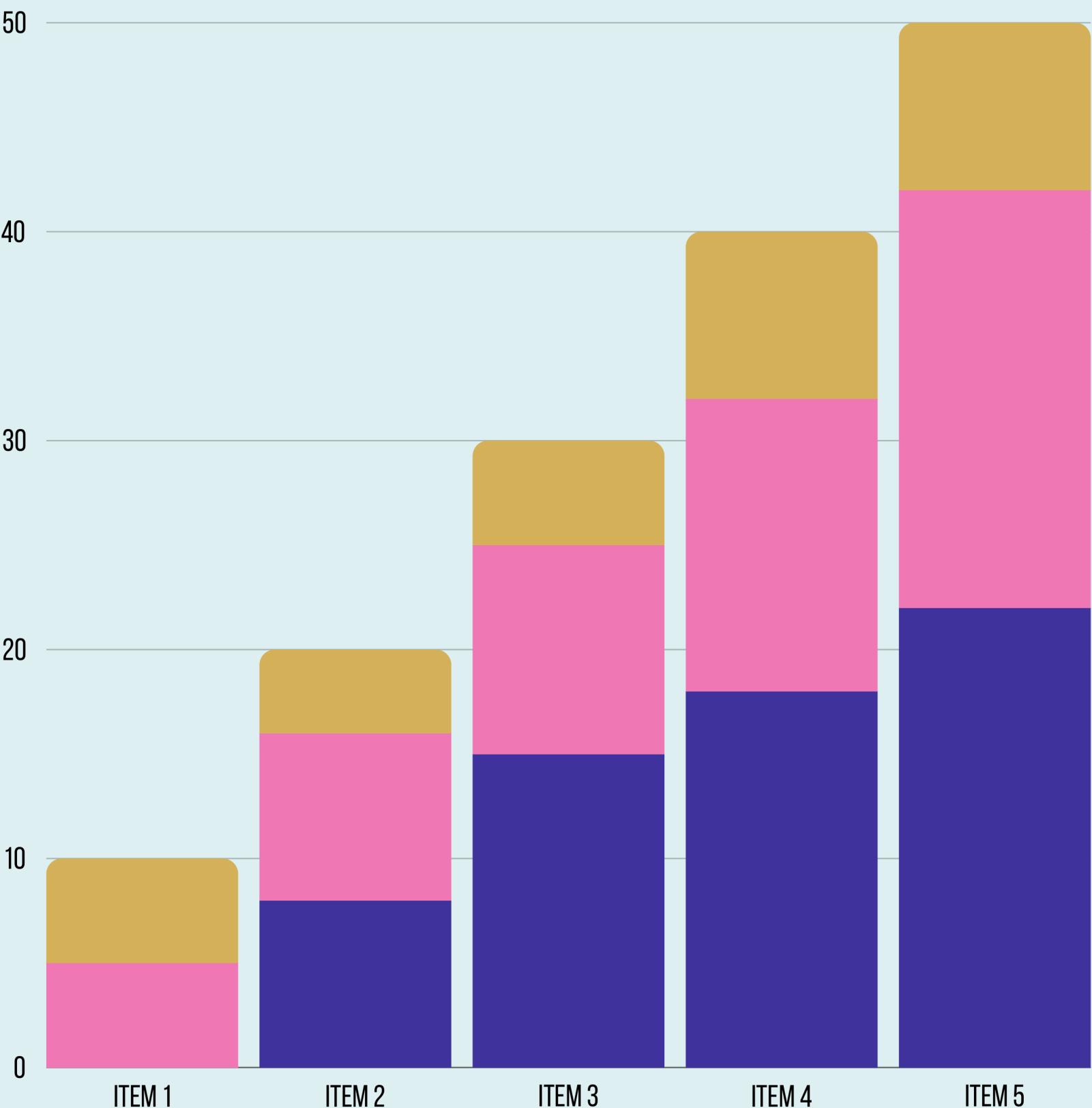
ANALIZE

PROBLEMI 😞



SKUP PODATAKA

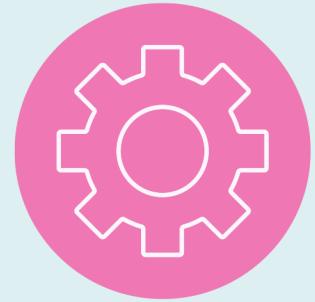
This Online Retail II data set contains all the transactions occurring for a UK-based and registered, non-store online retail between 01/12/2009 and 09/12/2011. The company mainly sells unique all-occasion gift-ware. Many customers of the company are wholesalers.





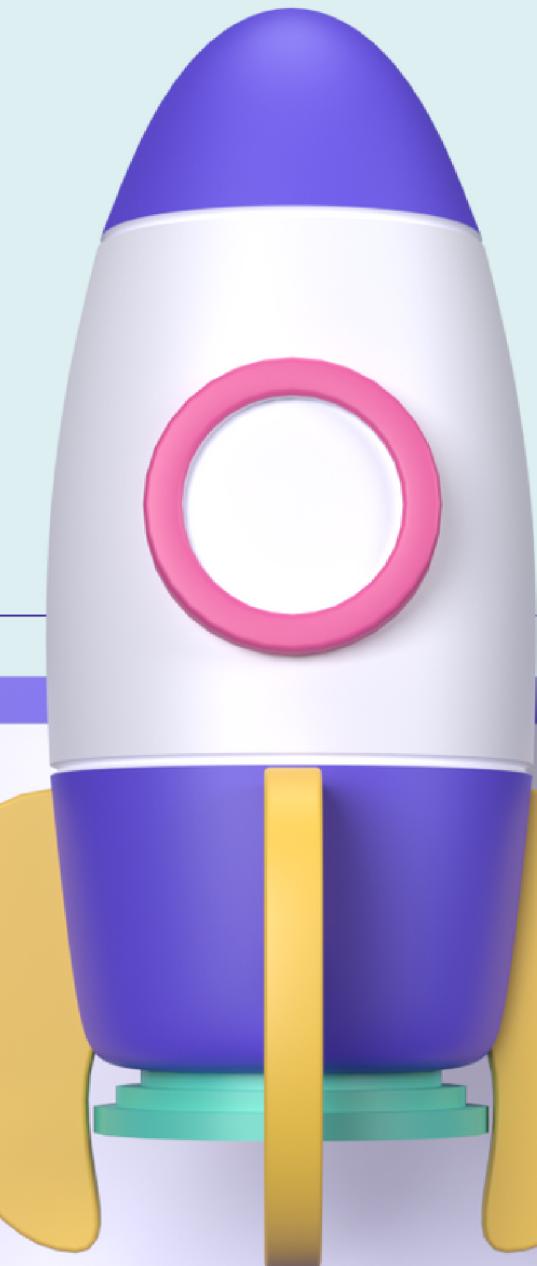
STEP ONE

Učitavanje



STEP TWO

Ispitivanje podataka



STEP THREE

Manipulacija podacima



STEP FOUR

Modeli

```
df.dtypes
```

Invoice	object
StockCode	object
Description	object
Quantity	int64
InvoiceDate	object
Price	float64
Customer ID	float64
Country	object
dtype:	object

```
df['StockCode'].unique()
```

```
array(['85048', '79323P', '79323W', ..., '23609', '23617', '23843'],  
      dtype=object)
```

```
df['StockCode'].nunique()
```

```
5305
```

```
df['Description'].unique()
```

```
array(['15CM CHRISTMAS GLASS BALL 20 LIGHTS', 'PINK CHERRY LIGHTS',  
      ' WHITE CHERRY LIGHTS', ..., 'mixed up',  
      'CREAM HANGING HEART T-LIGHT HOLDER',  
      'PAPER CRAFT , LITTLE BIRDIE'], dtype=object)
```

```
df['Description'].nunique()
```

```
5698
```

```
df.head()
```

Python

	Invoice	StockCode	Description	Quantity	InvoiceDate	Price	Customer ID	Country
0	489434	85048	15CM CHRISTMAS GLASS BALL 20 LIGHTS	12	2009-12-01 07:45:00	6.95	13085.0	United Kingdom
1	489434	79323P	PINK CHERRY LIGHTS	12	2009-12-01 07:45:00	6.75	13085.0	United Kingdom
2	489434	79323W	WHITE CHERRY LIGHTS	12	2009-12-01 07:45:00	6.75	13085.0	United Kingdom
3	489434	22041	RECORD FRAME 7" SINGLE SIZE	48	2009-12-01 07:45:00	2.10	13085.0	United Kingdom
4	489434	21232	STRAWBERRY CERAMIC TRINKET BOX	24	2009-12-01 07:45:00	1.25	13085.0	United Kingdom

```
df.duplicated().sum()
```

```
34335
```

```
df.drop_duplicates(inplace = True)
```

```
0
```

```
df.duplicated().sum()
```

```
df.isnull().sum()
```

Invoice	0
StockCode	0
Description	4275
Quantity	0
InvoiceDate	0
Price	0
Customer ID	0
Country	0

dtype: int64

```
df.dropna(inplace = True)
```

IZBACIVANJE DUPLIKATA I NULL VREDNOSTI

```
#Formiranje TotalPrice labele
```

```
df['TotalPrice'] = df['Quantity'] * df['Price']
```

```
df[df['TotalPrice']<=0].count()
```

TotalPrice - cena svake transakcije

DODAVANJE YEAR/MONTH KOLONE

```
df['year_month'] = df['Year'].astype(str) + '-' + df['Month'].astype(str)
```

```
# Convert 'year_month' to datetime
```

```
df['year_month'] = pd.to_datetime(df['year_month'], format='%Y-%m')
```

```
df.head()
```

	Invoice	StockCode	Description	Quantity	InvoiceDate	Price	Customer ID	Country	TotalPrice	Day	Month	Year	year_month
0	489434	85048	15CM CHRISTMAS GLASS BALL 20 LIGHTS	12	2009-12-01 07:45:00	6.95	13085.0	United Kingdom	83.4	1	12	2009	2009-12-01
1	489434	79323P	PINK CHERRY LIGHTS	12	2009-12-01 07:45:00	6.75	13085.0	United Kingdom	81.0	1	12	2009	2009-12-01
2	489434	79323W	WHITE CHERRY LIGHTS	12	2009-12-01 07:45:00	6.75	13085.0	United Kingdom	81.0	1	12	2009	2009-12-01
3	489434	22041	RECORD FRAME 7" SINGLE SIZE	48	2009-12-01 07:45:00	2.10	13085.0	United Kingdom	100.8	1	12	2009	2009-12-01
4	489434	21232	STRAWBERRY CERAMIC TRINKET BOX	24	2009-12-01 07:45:00	1.25	13085.0	United Kingdom	30.0	1	12	2009	2009-12-01

```
df=df.set_index('year_month')
```

SKRAĆIVANJE DATA SETA

```
proizvodi=df['stockCode'].unique()
type(proizvodi)
duzina=len(proizvodi)//3
pola_proizvoda=proizvodi[:duzina]
df = df[df['stockCode'].isin(pola_proizvoda)]
```

W

H

O

L

E

S

A

L

E

R

S

```
# Izračunajte 80% kvantil za TotalPrice  
threshold_amount = df['TotalPrice'].quantile(0.8)  
  
# Pronadite kupce čije su ukupne kupovine veće od 80% kvantila  
wholesale_customers = df.groupby('Customer ID')['TotalPrice'].sum()  
wholesale_customers = wholesale_customers[wholesale_customers > threshold_amount]
```

#Analiza koliko često obavljaju takve kupovine:

```
# Pretvorite InvoiceDate u datetime format  
df['InvoiceDate'] = pd.to_datetime(df['InvoiceDate'])  
  
# Grupišite po kupcima i analizirajte vremenski raspon između transakcija  
wholesale_purchase_frequency = df[df['Customer ID'].isin(wholesale_customers.index)]  
wholesale_purchase_frequency = wholesale_purchase_frequency.groupby('Customer ID')['InvoiceDate'].diff().mean()  
  
wholesale_purchase_frequency
```

Timedelta('1 days 18:17:44.074336911')

#Pronalaženje redovnih kupaca:

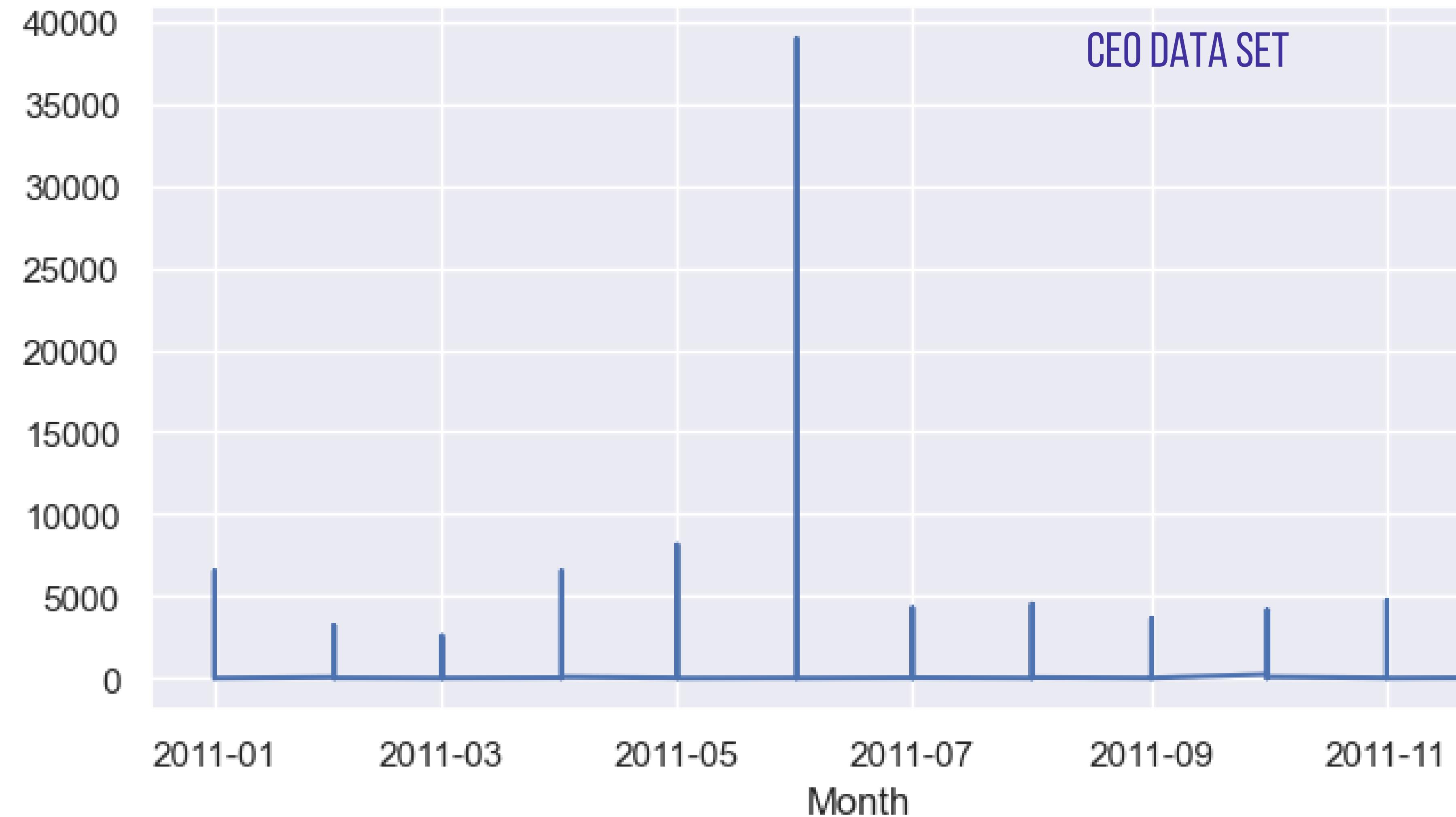
```
transaction_counts = df['Customer ID'].value_counts()  
print(transaction_counts.describe())
```

count	4153.000000
mean	66.156032
std	1077.240394
min	1.000000
25%	9.000000
50%	22.000000
75%	54.000000
max	69018.000000
Name:	count, dtype: float64

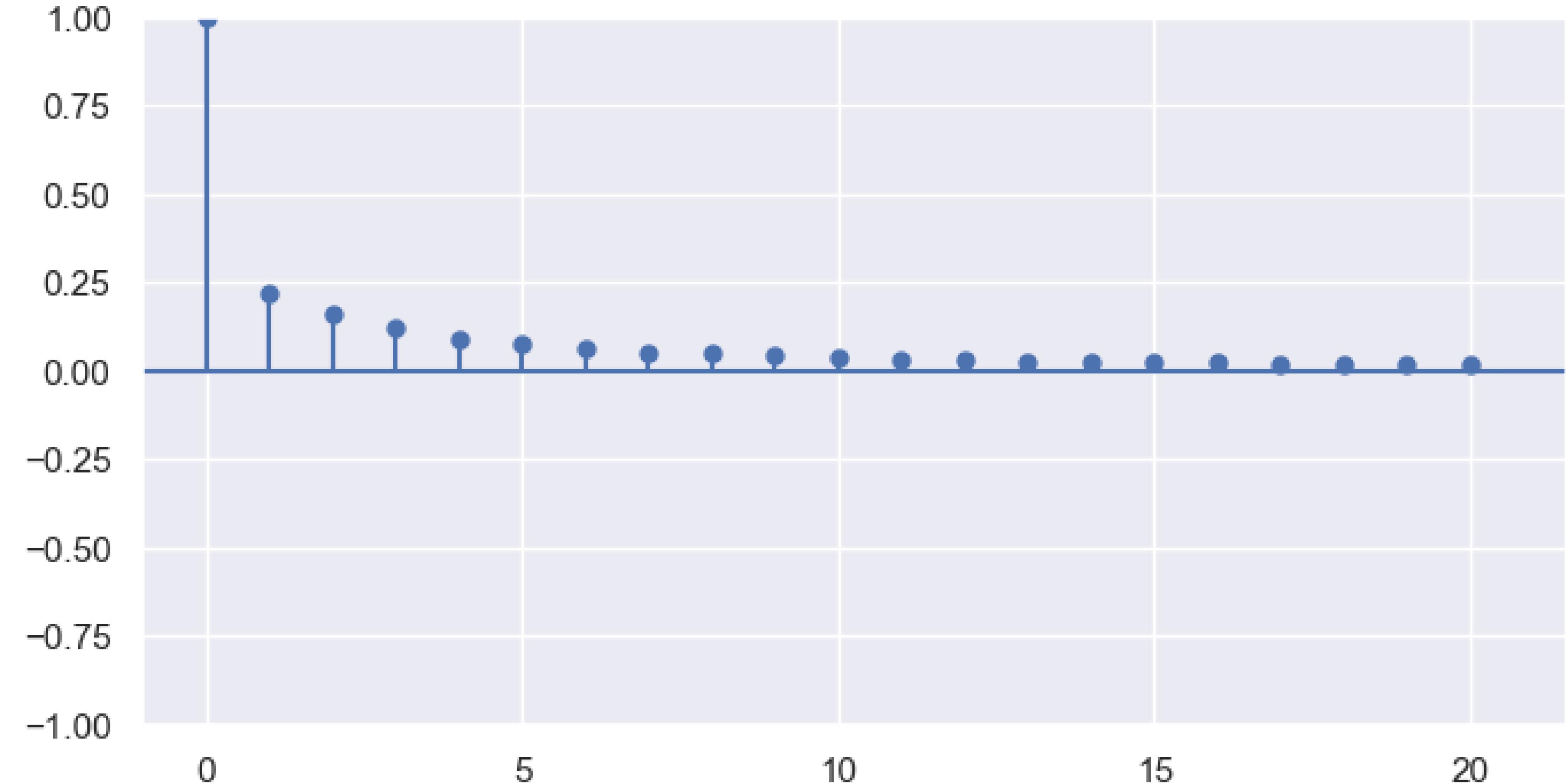
#Analiza onih koji su kupovali samo jednom:

```
one_time_customers = df['Customer ID'].value_counts()  
one_time_customers = one_time_customers[one_time_customers == 1]
```

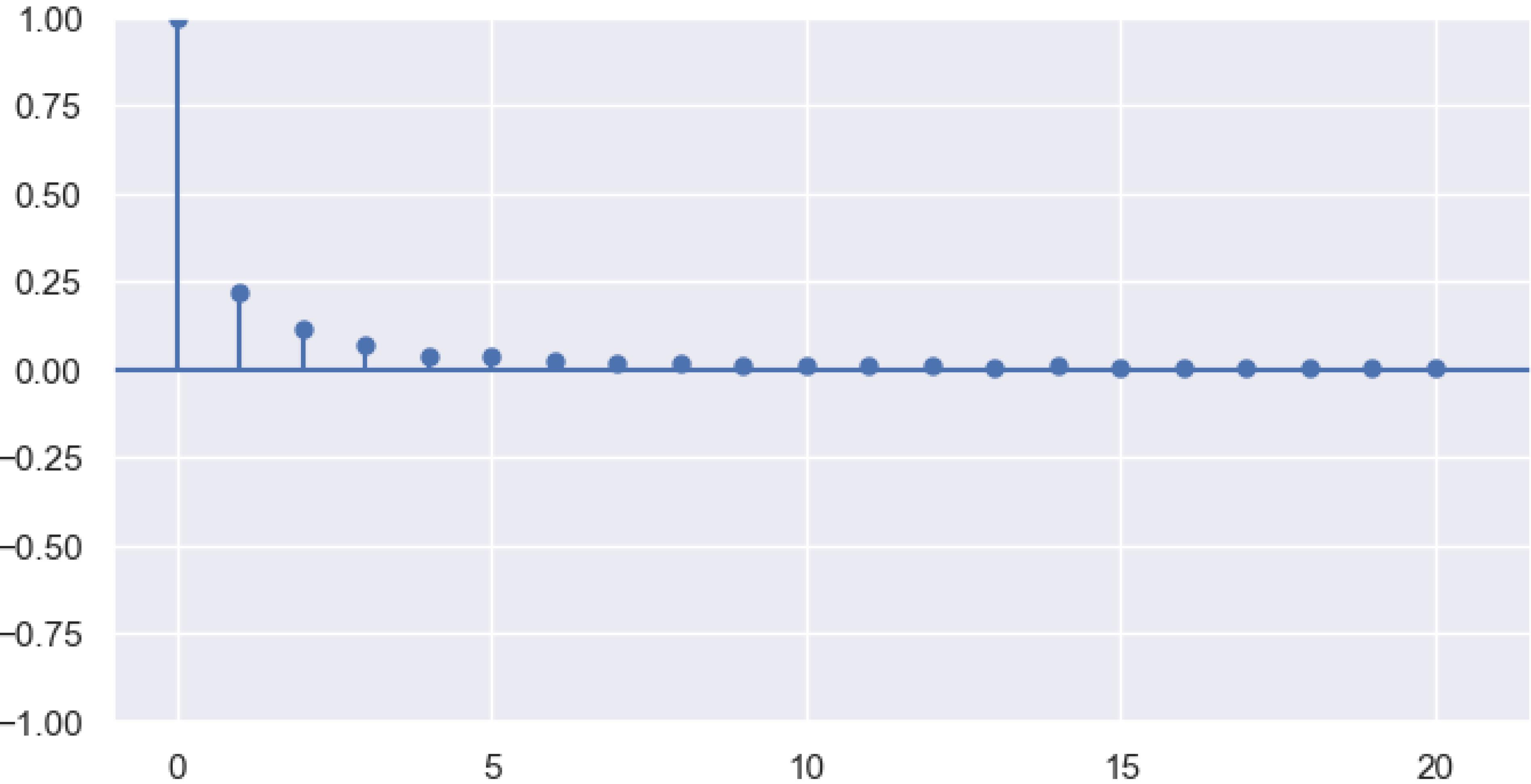
CEO DATA SET



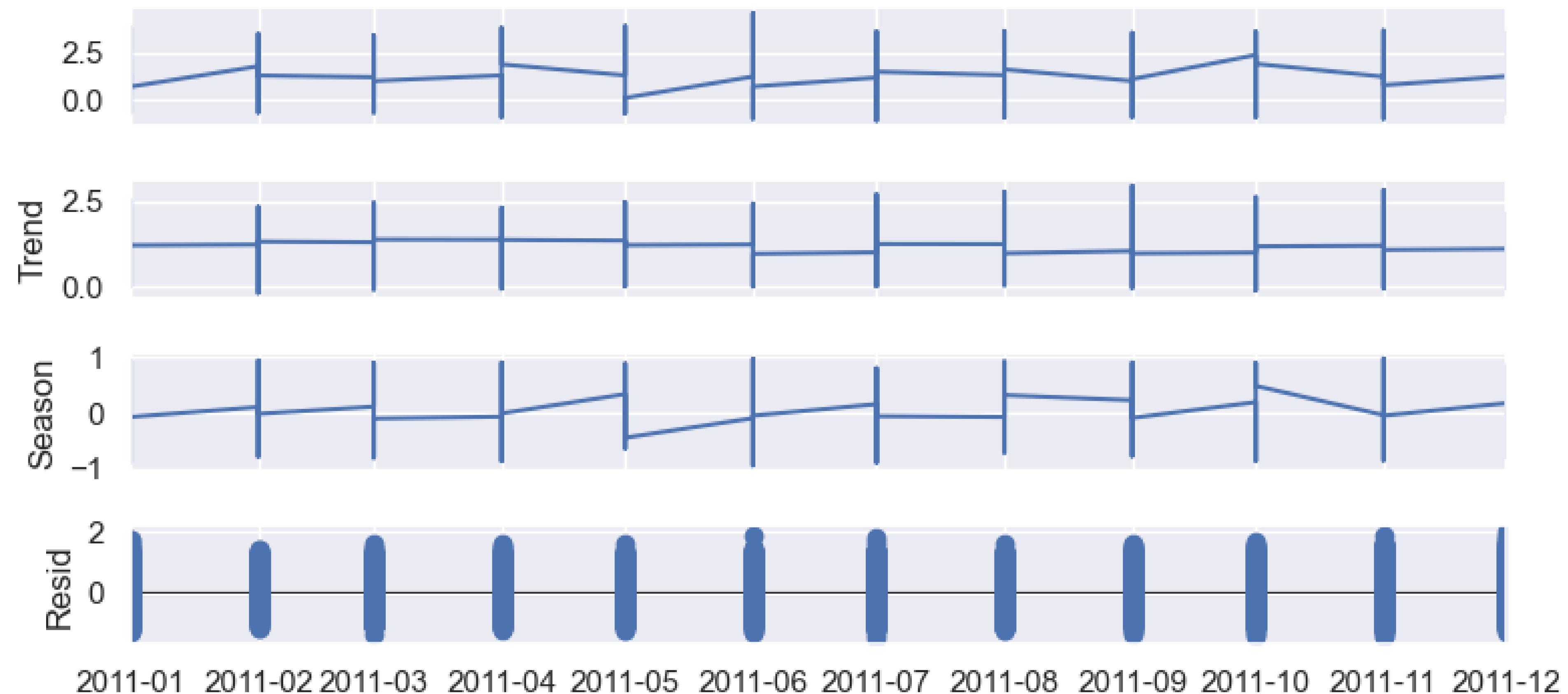
Autocorrelation



Partial Autocorrelation



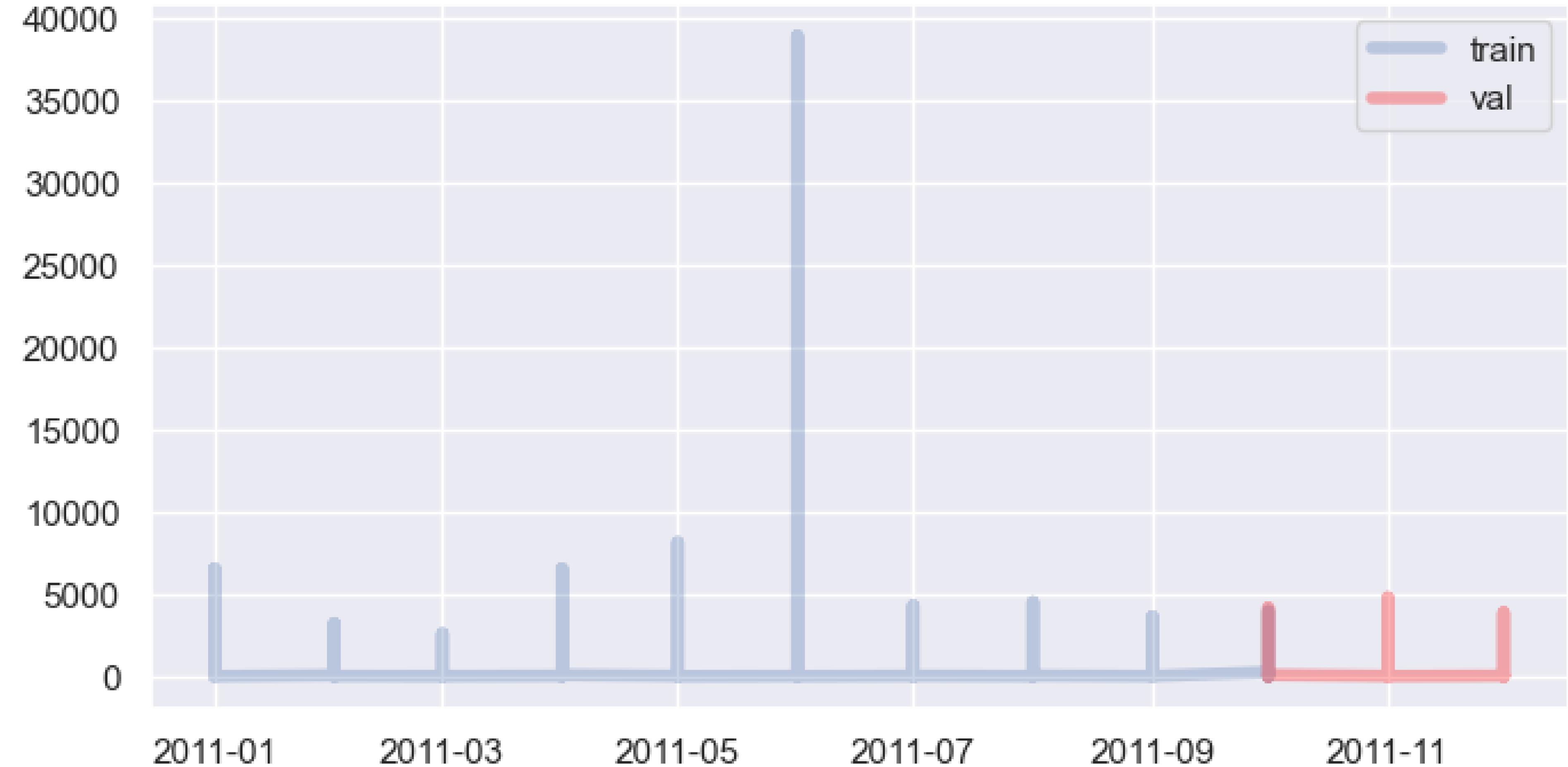
$\log_{10}(\text{TotalPrice})$



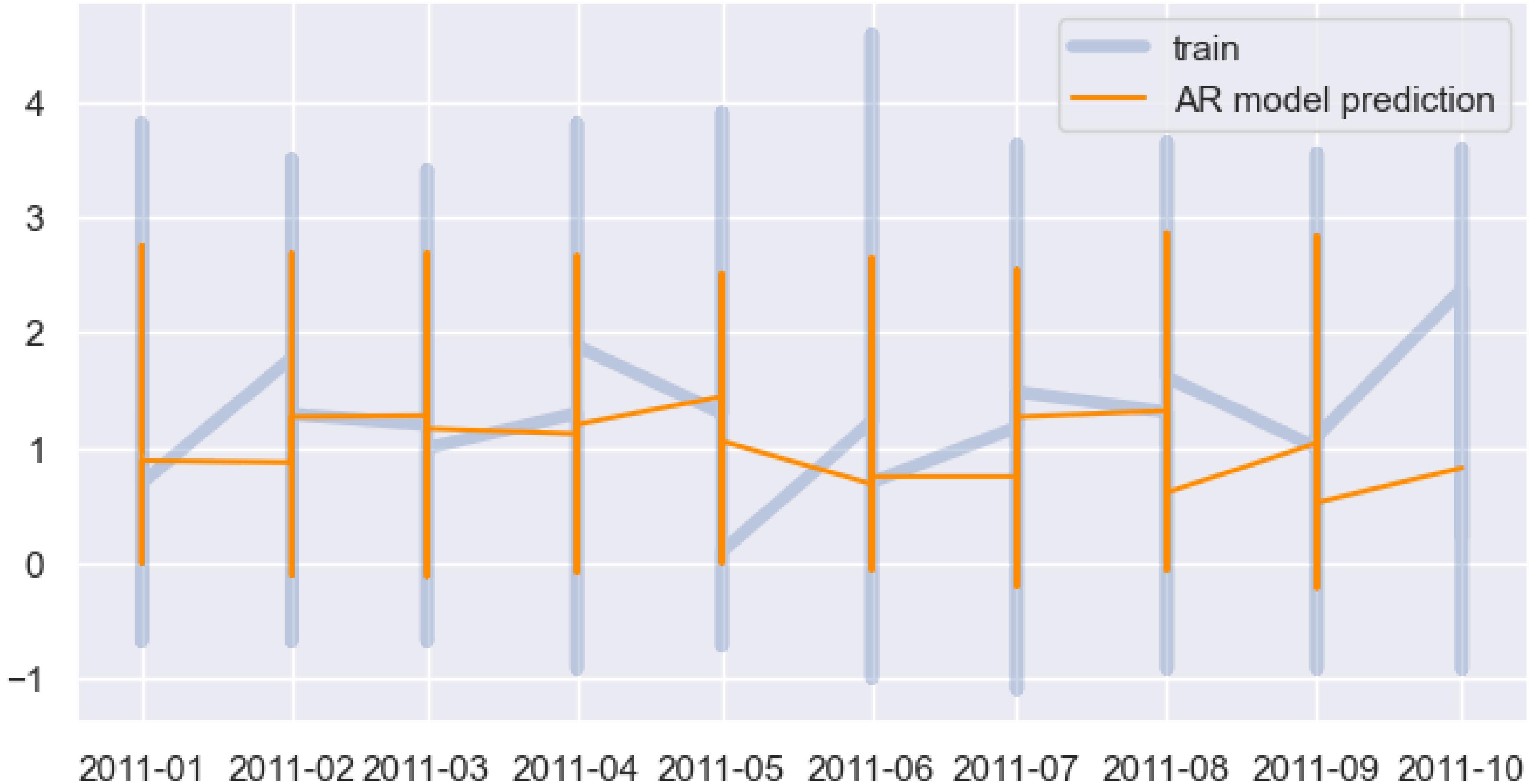
```
df['log10(TotalPrice)'].plot();
from statsmodels.tsa.stattools import adfuller
adf_value = adfuller(df['log10(TotalPrice)'])[0]
p_value = adfuller(df['log10(TotalPrice)'])[1]
print(f'{adf_value:.2f}, {p_value:.2f}')

if p_value <= 0.05: print('postoji stacionarnost')
else: print('ne postoji stacionarnost')
```

```
adf_value=-42.08, p_value=0.00
postoji stacionarnost
```



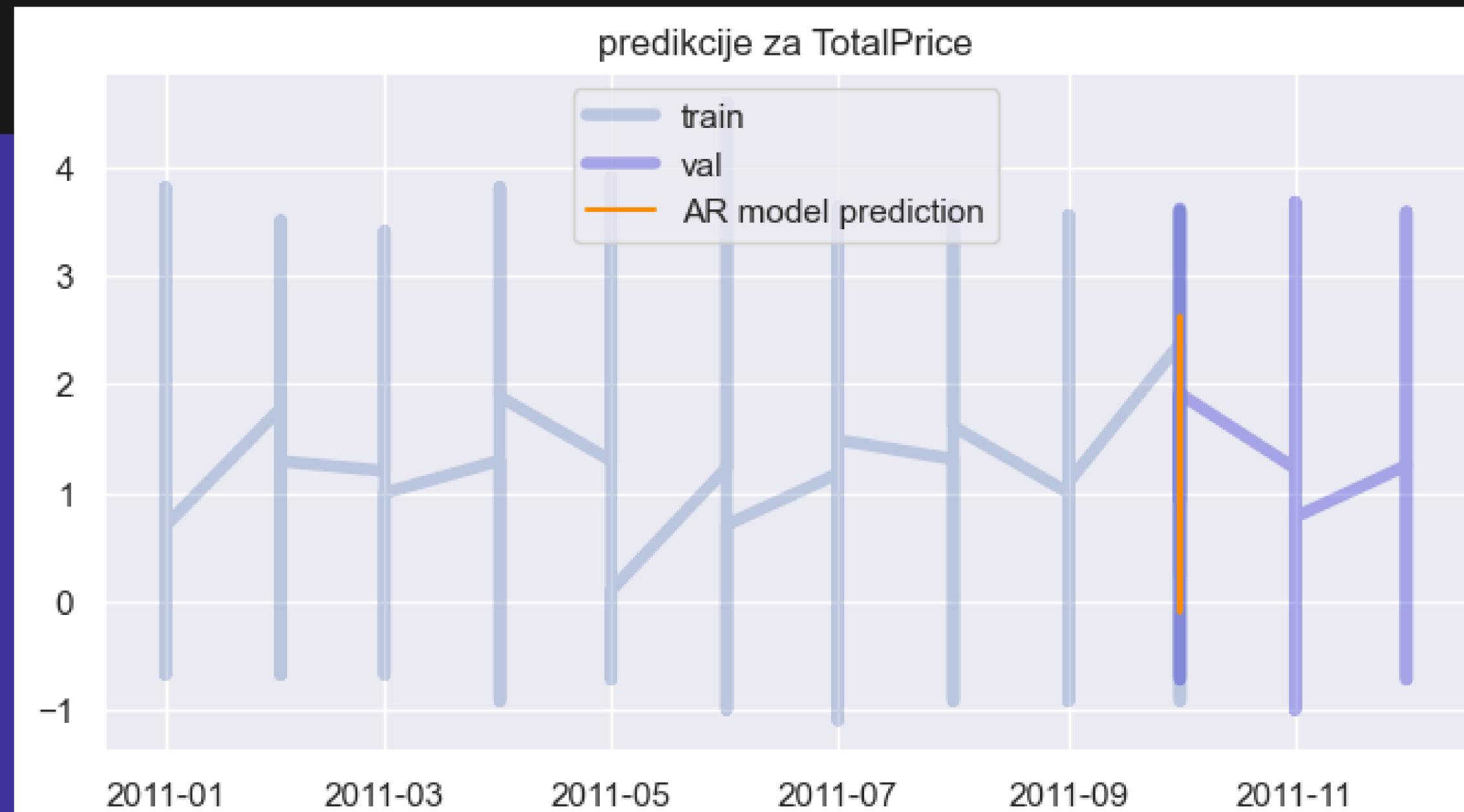
predikcije za log10(TotalPrice)



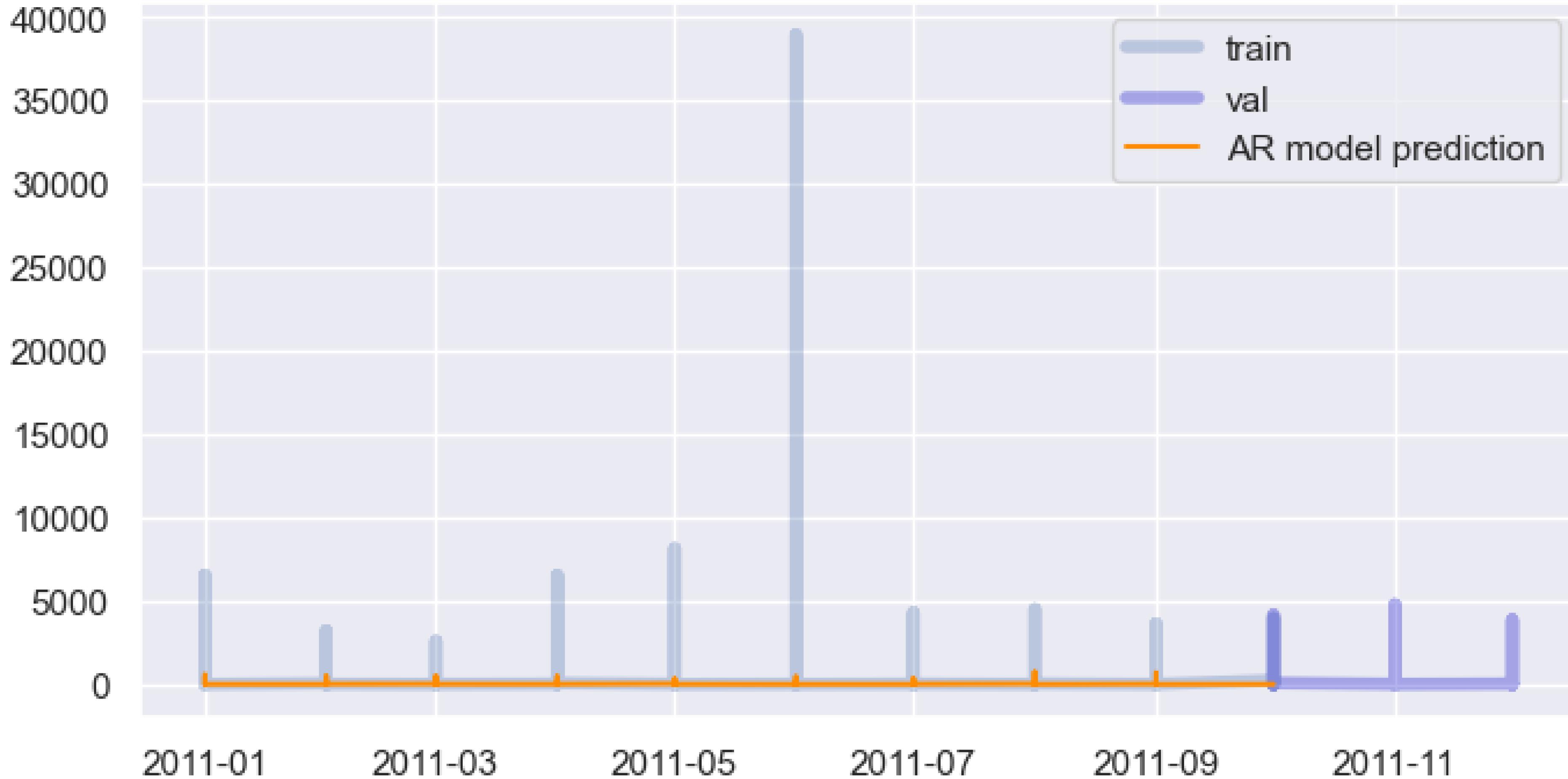
```
val_df['log10(TotalPrice)'] = np.log10(val_df['TotalPrice'])
y_val_pred = ar_model.predict(start=val_df.index[0])

plt.plot(train_df['log10(TotalPrice)'], color='b', linewidth=4, alpha=0.3, label='train')
plt.plot(val_df['log10(TotalPrice)'], color='mediumblue', linewidth=4, alpha=0.3, label='val')

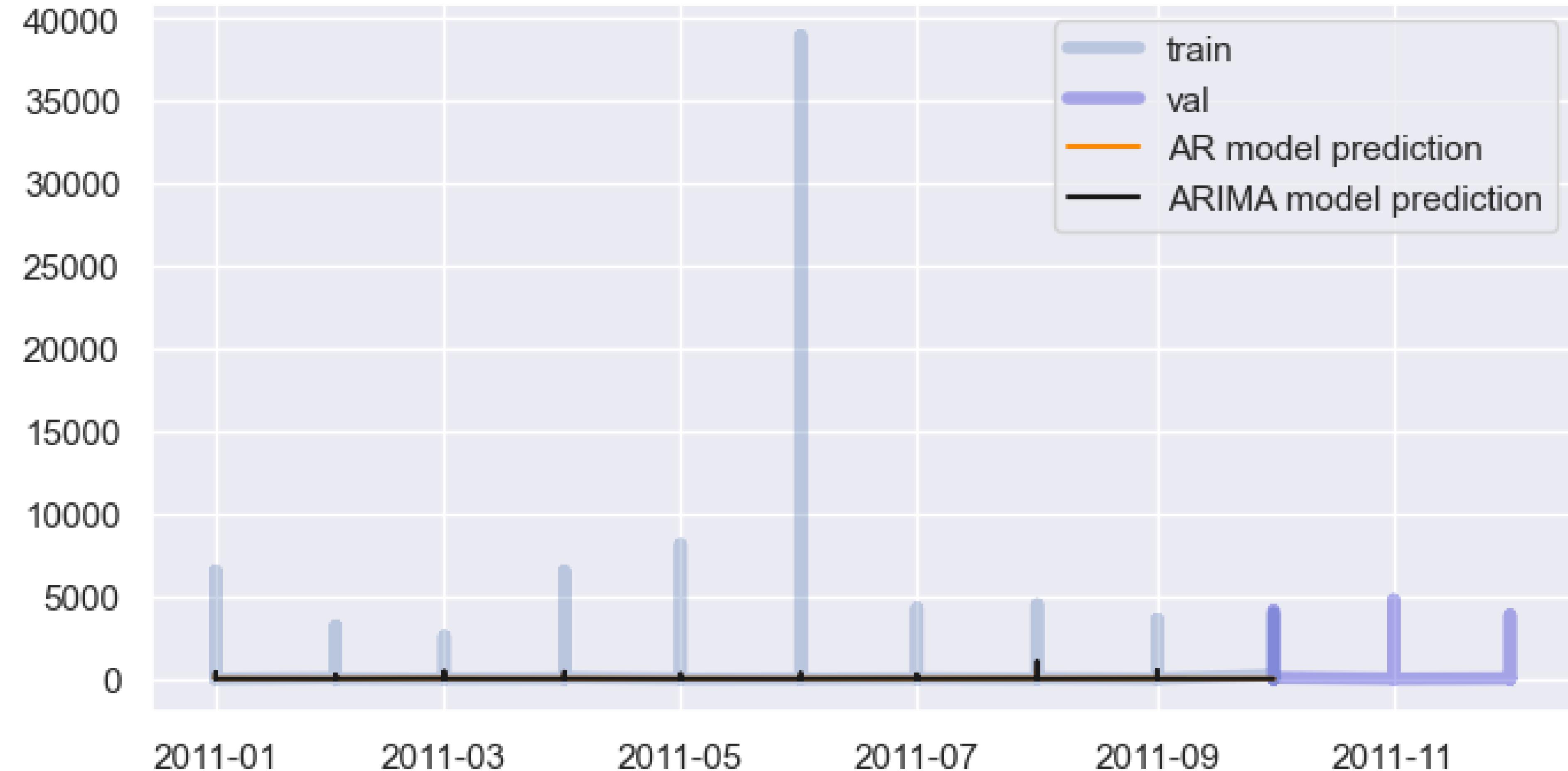
plt.plot(y_val_pred, color='darkorange', label='AR model prediction')
plt.title('predikcije za TotalPrice')
plt.legend()
plt.show()
```



predikcije za TotalPrice



predikcije za TotalPrice



AR model evaluacija nad validacionim skupom:

$\text{mse}(\text{actual}, \text{ar_pred})=9304.11$ $\text{mae}(\text{actual}, \text{ar_pred})=18.21$

ARIMA model evaluacija nad validacionim skupom:

$\text{mse}(\text{actual}, \text{arima_pred})=9575.12$ $\text{mae}(\text{actual}, \text{arima_pred})=19.26$

PROBLEMI

MEMORYERROR



```
df.shape  
(1067371, 8)
```

```
import gc  
gc.collect()
```



ZAKLJUČAK



HVALA NA PAŽNJI

