

**Прогнозирование временных рядов на основе использования
разработанной в среде R Автоматизированной системы
прогнозирования и анализа временных рядов (АСПАВР),
а также применения технологии глубокого обучения для RNN
LSTM в среде MatLab.**

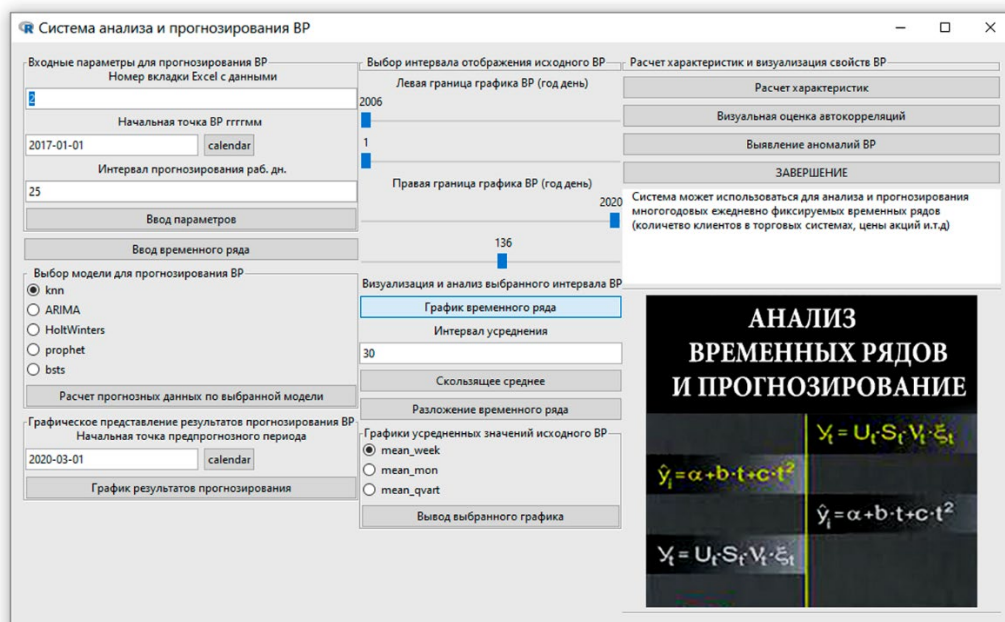
Нами была разработана в среде R система, позволяющая автоматизировать процесс анализа временных рядов и получения прогноза их развития. В систему интегрированы современные методы анализа и прогнозирования ВР.

АСПАВР реализует следующие основные функции:

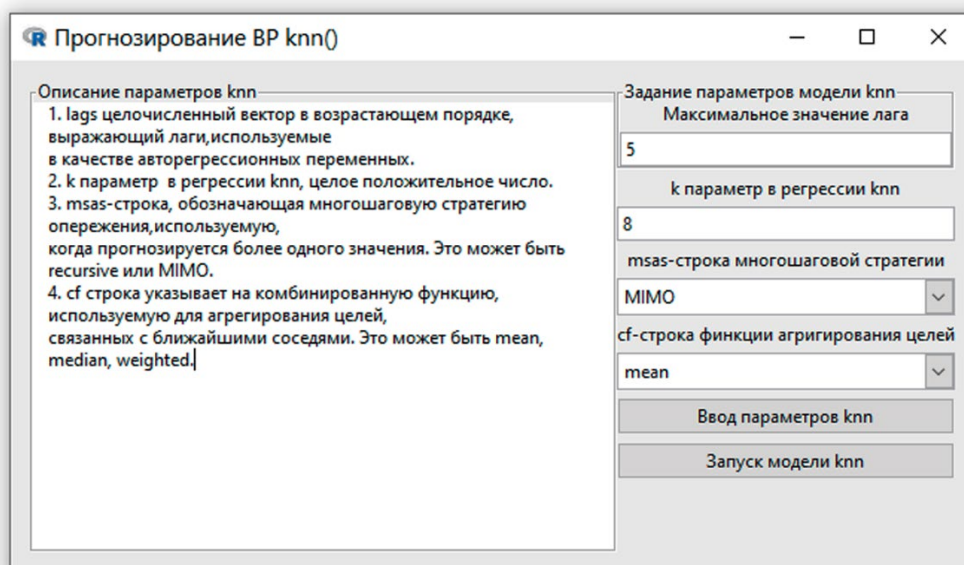
- позволяет вводить начальную временную точку ВР и задавать период прогнозирования;
- загружать ВР из таблицы Excel и преобразовывать его в необходимые для анализа и прогнозы формы;
- выбирать метод прогнозирования и получать результаты в числовой и графической форме. В системе реализованы следующие методы: knn (ближайших соседей), ARIMA (интегрированная модель авторегрессии - скользящего среднего с поиском оптимальных параметров), HoltWinters (усовершенствованный метод экспоненциального сглаживания), prophet (метод пророк -это современный метод, разработанный компанией Facebook, позволяющий в автоматическом режиме создавать модели, обладающие высокой точностью предсказаний), bstс (модель байесовских структурных временных рядов).
- позволяет получать в графической и числовой форме консенсус прогноз на основе выше приведенных методов;
- выбирать произвольный интервал в пределах ВР и визуализировать его;
- задавать интервал усреднения и получать трансформированный ряд методом скользящего среднего;
- осуществлять разложение ВР на составляющие;
- усреднять ряд по неделям, месяцам и кварталам и давать соответствующие графические представления;
- рассчитывать несколько десятков характеристик ВР;
- осуществлять визуальную оценку автокорреляций ВР;
- выявлять аномалии в ВР.

Система создавалась для анализа и прогноза многолетних временных рядов с ежедневным временным шагом (биржевые цены акций, сырья, драгоценных металлов, котировки валют и т. д). В частности, она регулярно применяется для прогноза акций Газпрома и цены нефти Brent. В этом материале мы представим результаты прогноза цен закрытия акций Газпрома на Московской бирже на 25 рабочих дней с 1 июня 2021 года.

Основной экран системы имеет следующий вид:



Окно метода Knn:



Окна методов HoltWinters и Prophet

Прогнозирование BP HoltWinters()

Описание параметров HoltWinters

- 1.alpha- параметр, значения: NULL, TRUE
- 2.beta- параметр, значения: NULL, TRUE, FALSE; если FALSE, то выполняется экспоненциальное сглаживание.
- 3.gamma параметр, значения: NULL, TRUE, FALSE; если FALSE, то используется несезонная модель.
- 4.seasonal строка символов, значения add, mult для выбора мультипликативной или аддитивной модели.
- 5.start.periods начальные периоды, используемые при автоопределении начальных значений ≥ 2 .
6. optim.start -вектор с исходными для оптимизатора значениями alpha,beta,gamma.

Задание параметров модели HoltWinters

Параметр alpha

NULL

Параметр beta

NULL

Параметр gamma

NULL

Вид сезонной модели

additive

Число начальных периодов

2

Начальные значения параметров для оптимизатора

alpha

0.3

beta

0.1

gamma

0.1

Ввод параметров HoltWinters

Запуск модели HoltWinters

Прогнозирование BP prophet()

Описание параметров prophet

1. growth Строка 'linear' или 'logistic' для указания линейного или логистического тренда.
2. n.changepoints количество возможных точек изменения
3. changepoint.range доля истории, в которой будут оцениваться точки изменения тренда.
4. yearly.seasonality годовая сезонность. Может быть 'auto', TRUE, FALSE
5. weekly.seasonality еженедельная сезонность. Может быть 'auto', TRUE, FALSE
6. daily.seasonality ежедневная сезонность. Может быть 'auto', TRUE, FALSE
7. seasonality.mode вид модели: additive, multiplicative
8. seasonality.prior.scale параметр, моделирующий силу модели сезонности. Большие значения позволяют модели соответствовать большим сезонным колебаниям, меньшие значения ослабляют сезонность
9. holidays.prior.scale параметр, моделирующий силу модели компонентов выходных
10. changepoint.prior.scale параметр, моделирующий гибкость автоматического выбора точки переключения. Большие значения позволят много точек изменения, меньшие значения позволят мало точек изменения
11. mcmc.samples Целое число, если оно больше 0, будет делать полный байесовский вывод с указанным числом выборок MCMC. Если 0, будет делать оценку MAP.
12. interval.width Числовая, ширина интервалов неопределенности, предусмотренных для прогноза
13. uncertainty.samples Количество смоделированных розыгрышей, использованных для оценки интервалов неопределенности. Установка этого значения в 0 отключит оценку неопределенности и ускорит расчет.
14. fit Boolean, если 0, модель инициализируется, но не выполняется.
15. country_name код страны для учета праздников

Задание параметров модели prophet

Параметр growth

linear

Параметр n.changepoints

45

Параметр changepoint.range

0.85

Параметр yearly.seasonality

TRUE

Параметр weekly.seasonality

FALSE

Параметр daily.seasonality

FALSE

Параметр seasonality.mode

additive

Параметр seasonality.prior.scale

20

Задание параметров модели prophet

Параметр holidays.prior.scale

10

Параметр changepoint.prior.scale

0.08

Параметр mcmc.samples

0

Параметр interval.width

0.9

Параметр uncertainty.samples

100

Параметр fit Boolean

0

Параметр country_name

RU

Ввод параметров prophet

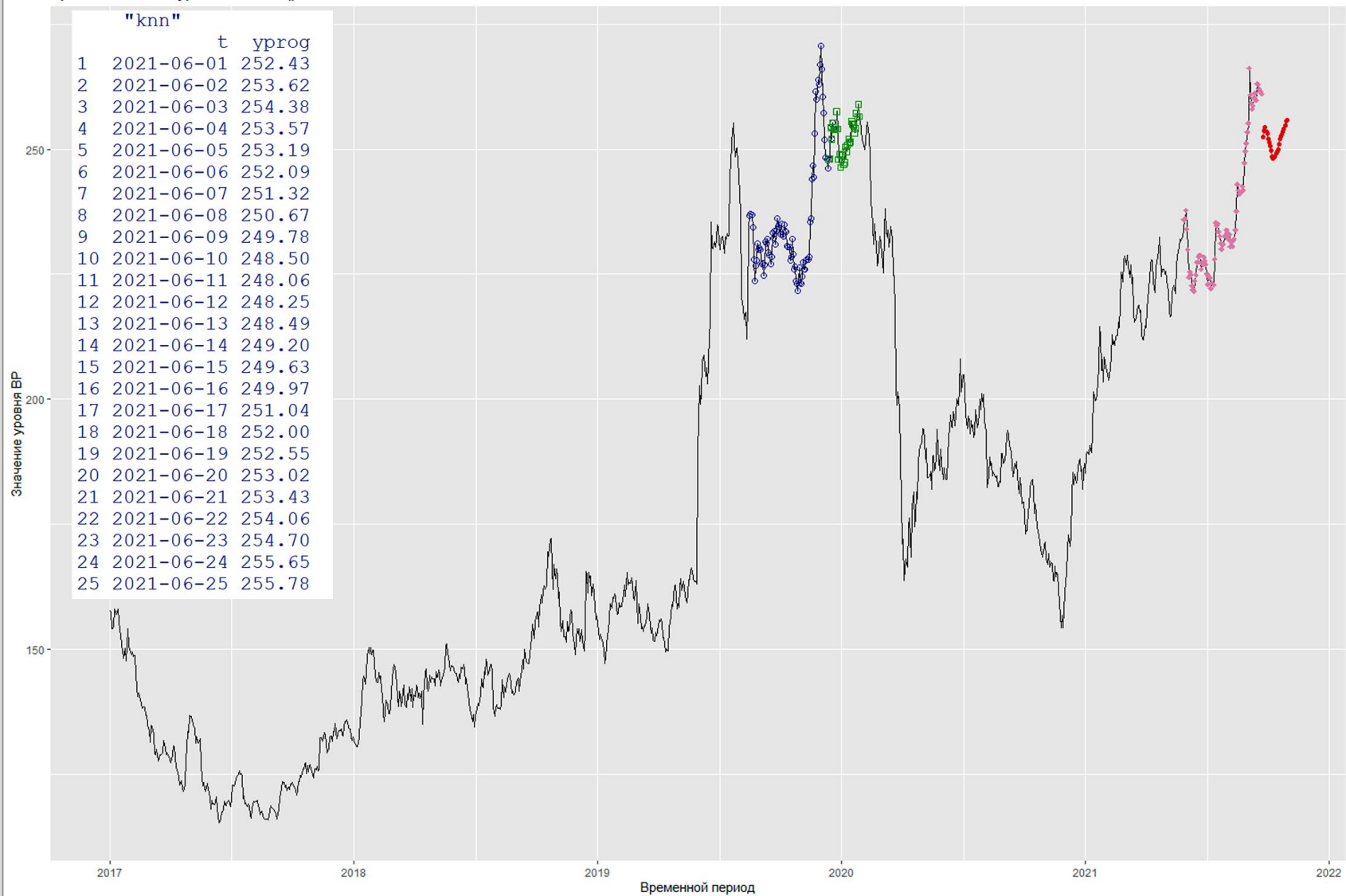
Запуск модели prophet

Параметры методов ARIMA и bsts определяются автоматически на основе определенных критериев.

Результаты прогнозирования и анализа представлены ниже:

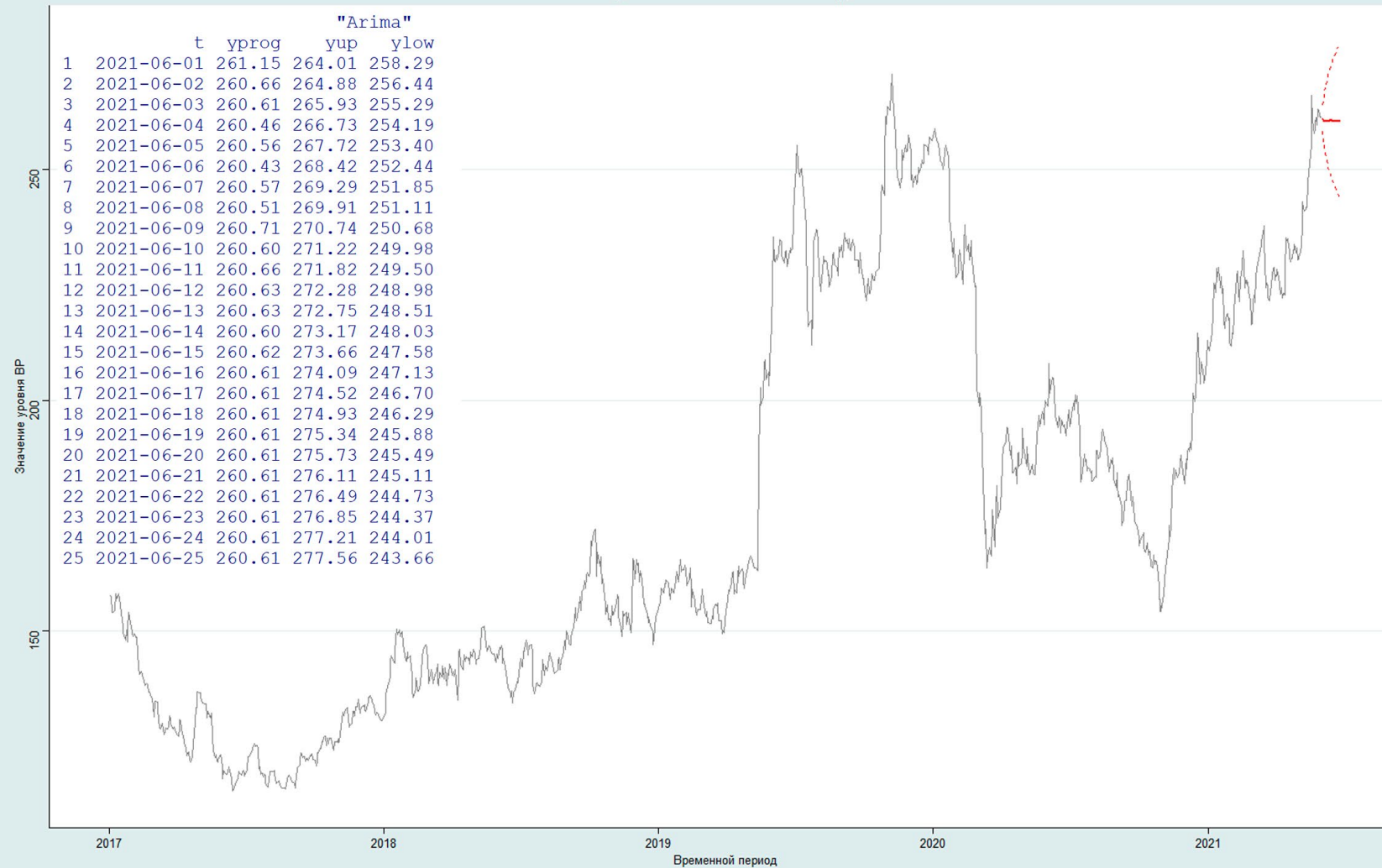


Прогноз значений уровней BP knn()



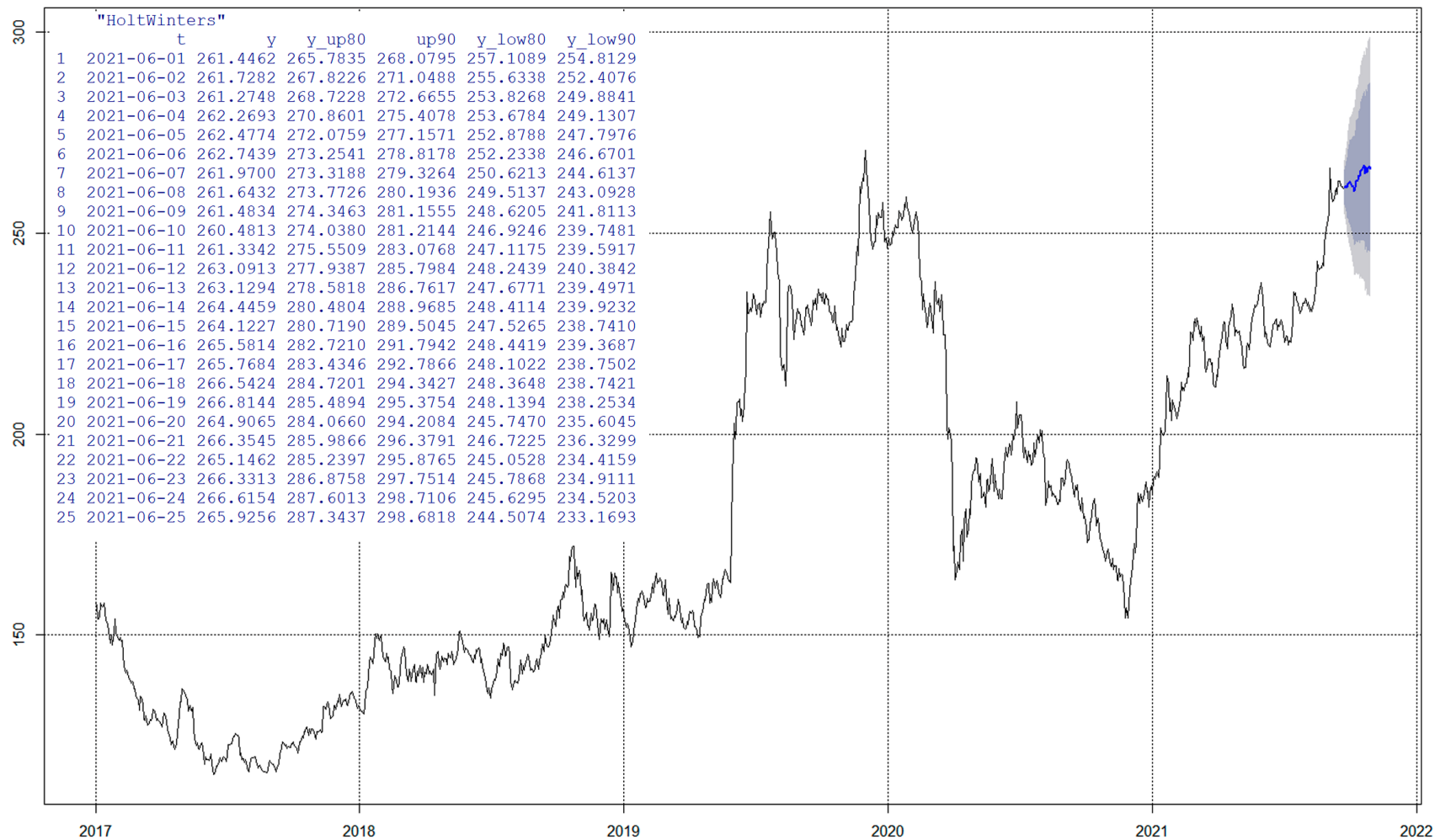


Прогноз значений BP ARIMA()



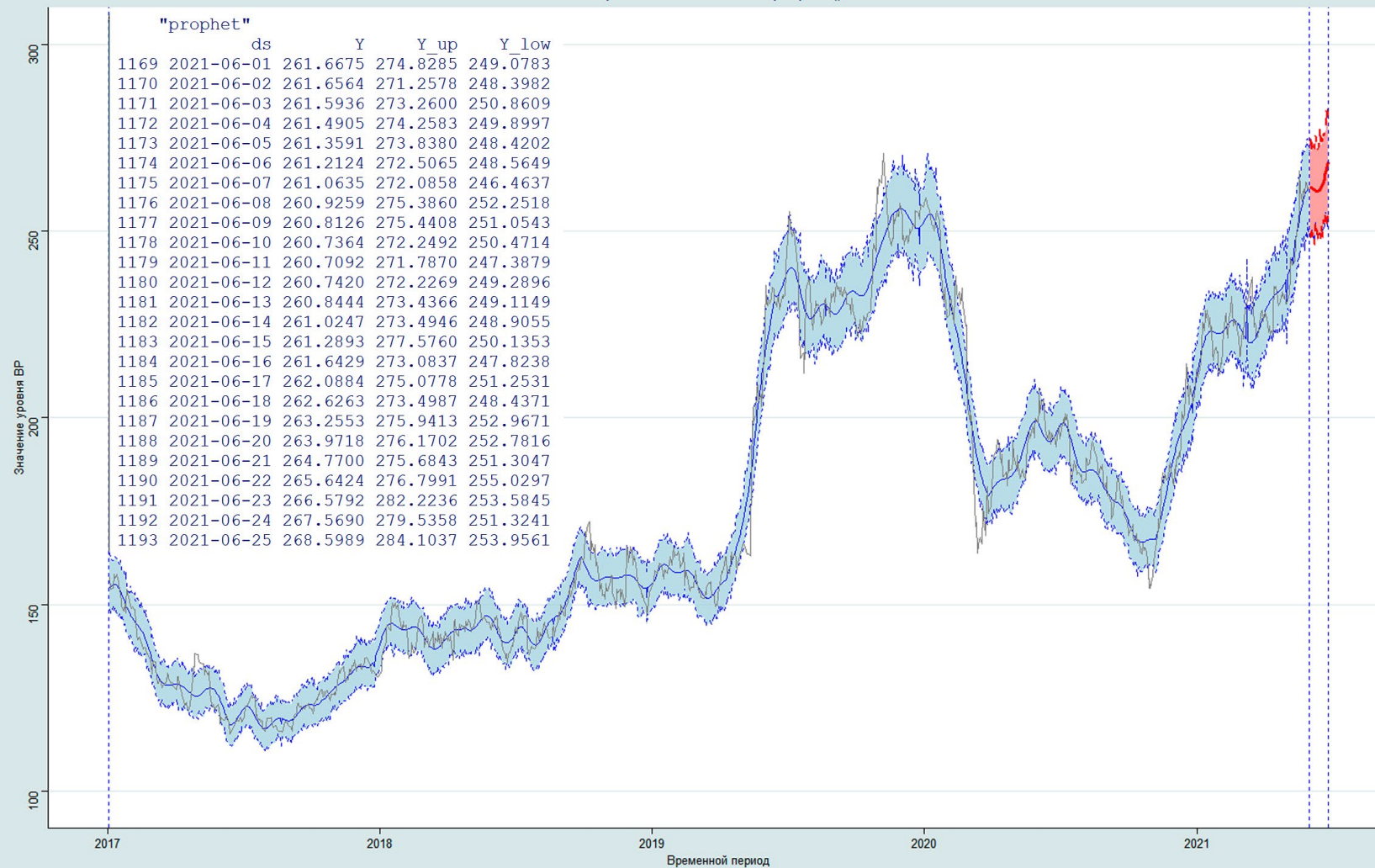


Прогноз значений BP HoltWinters()



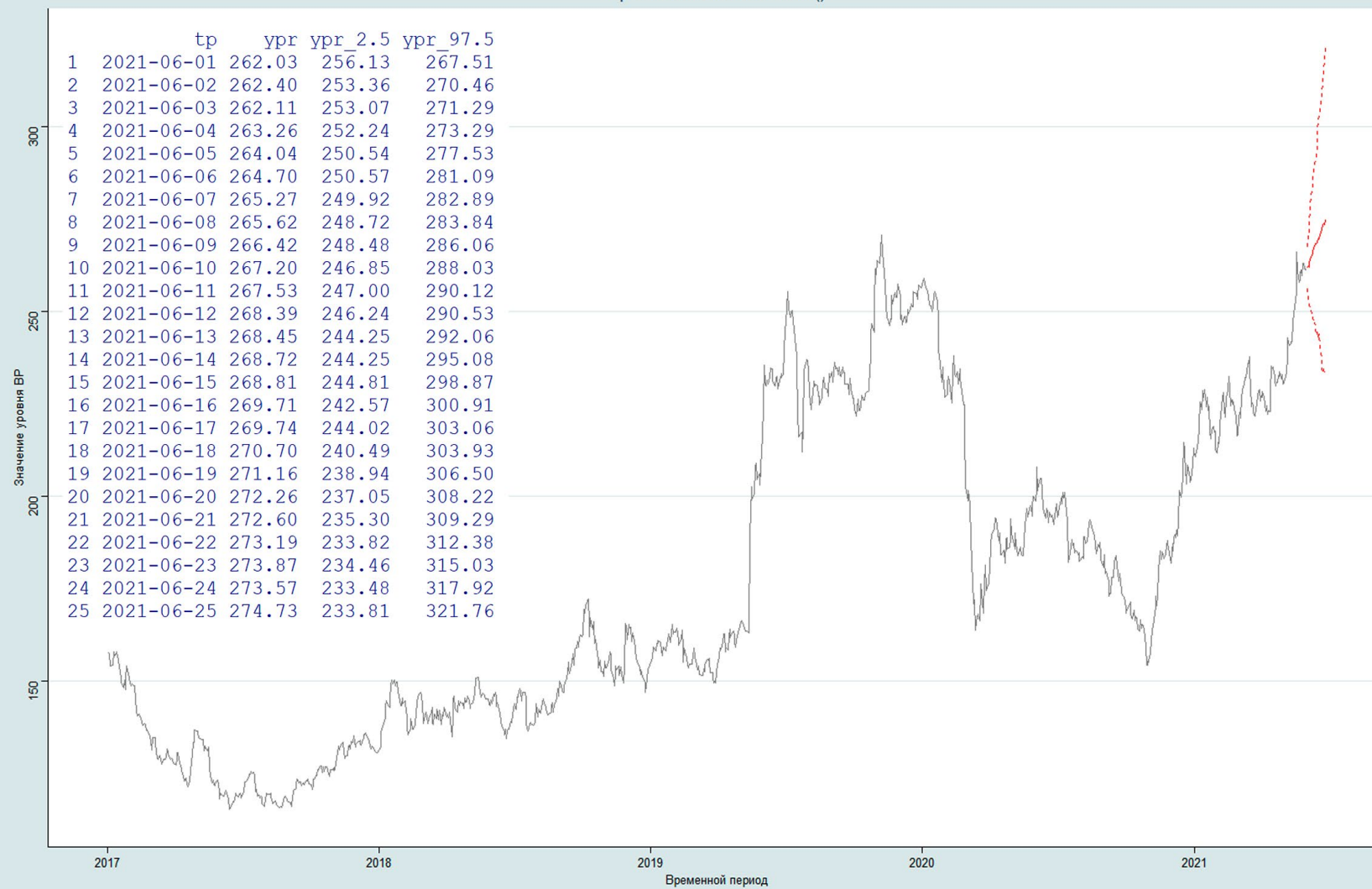


Прогноз значений BP prophet()





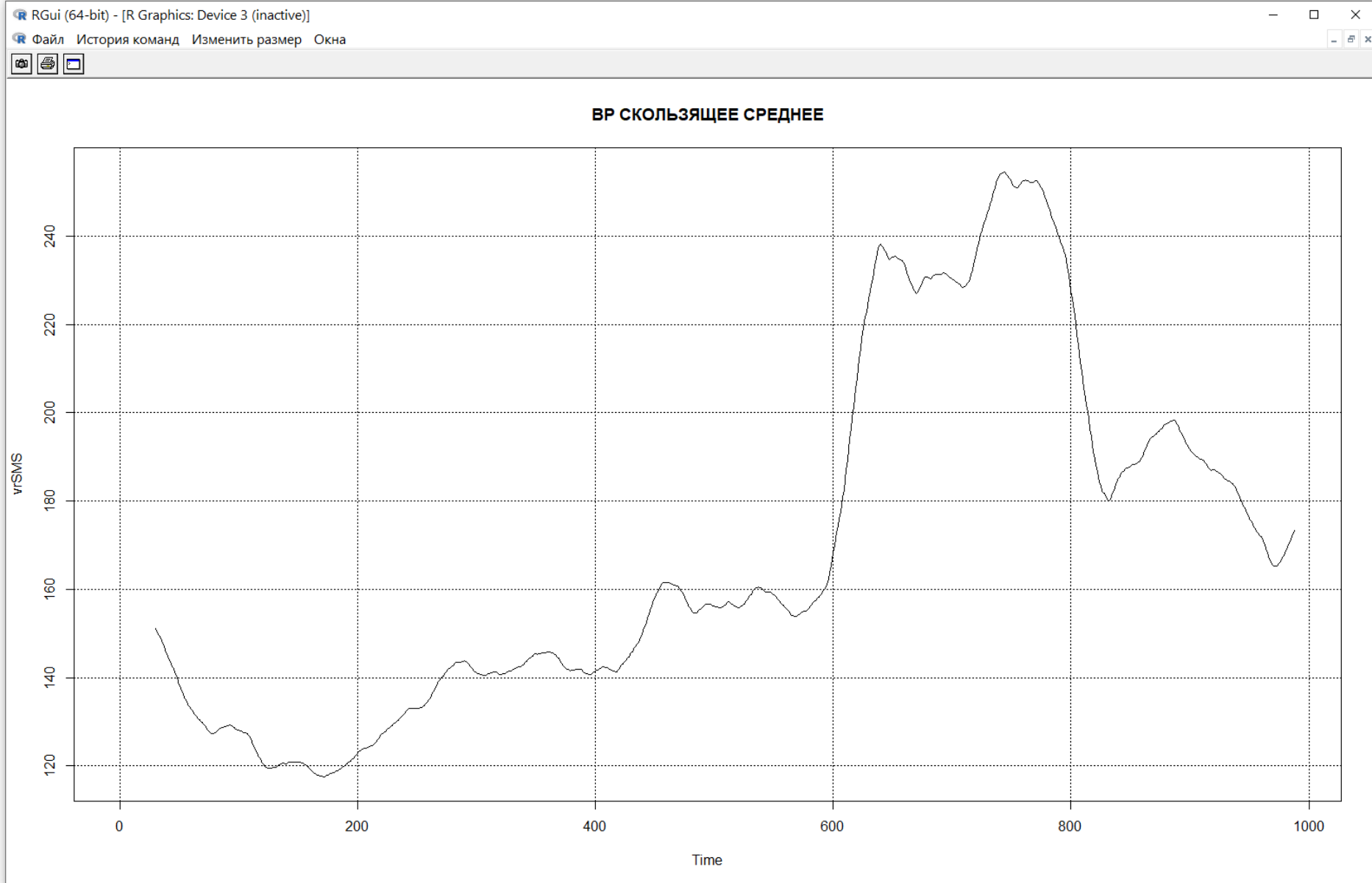
Прогноз значений BP bstс()

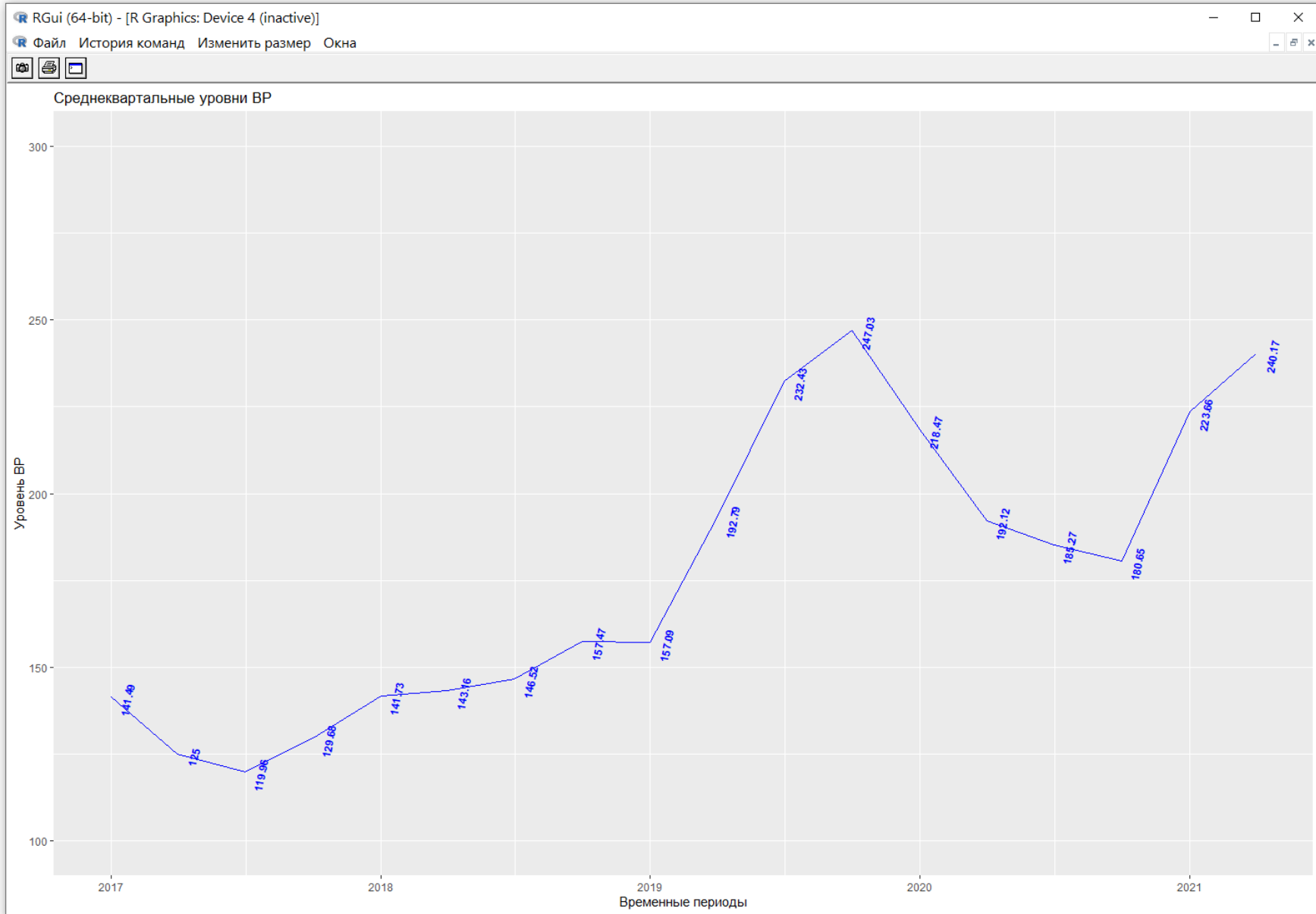


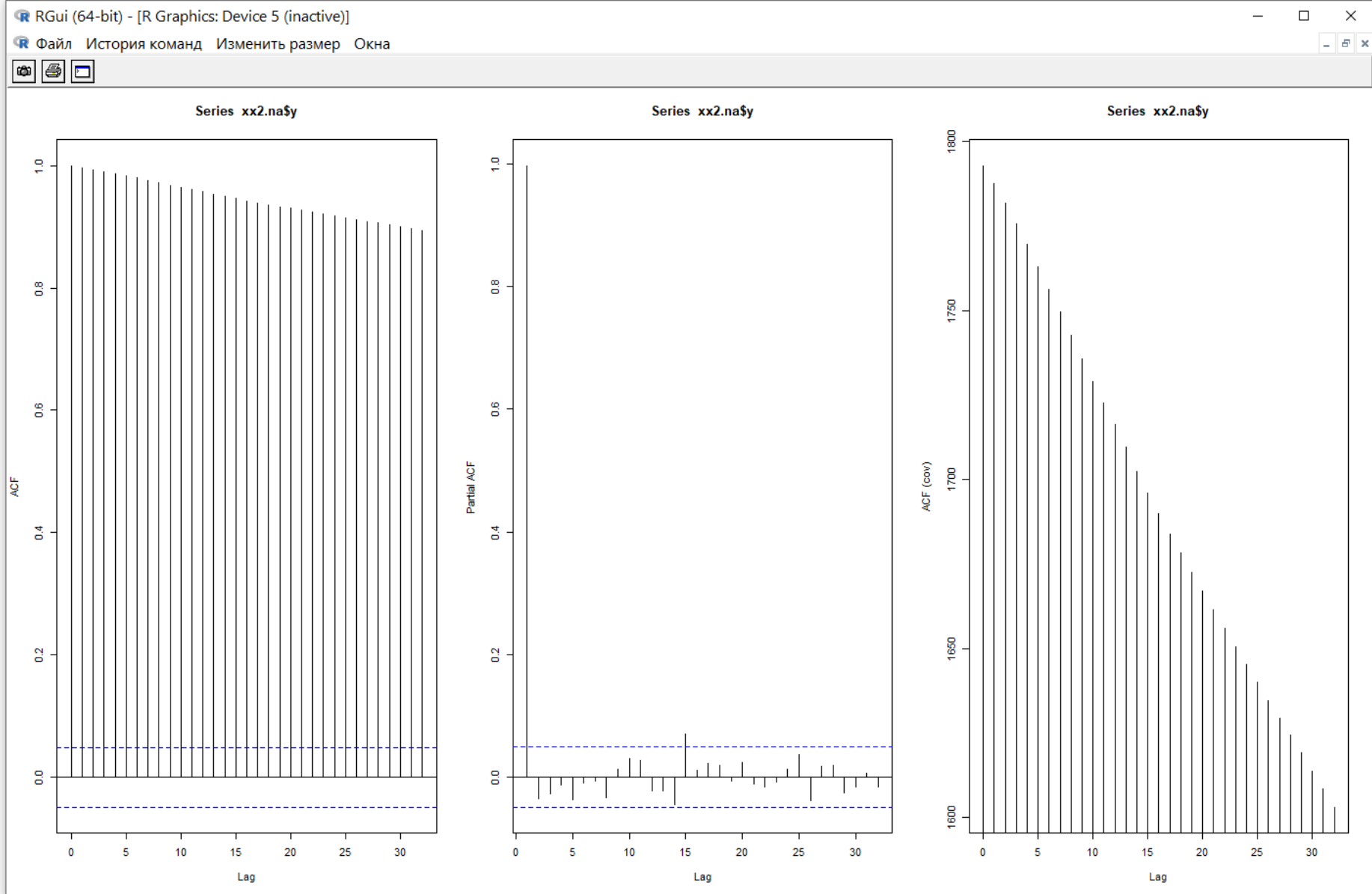


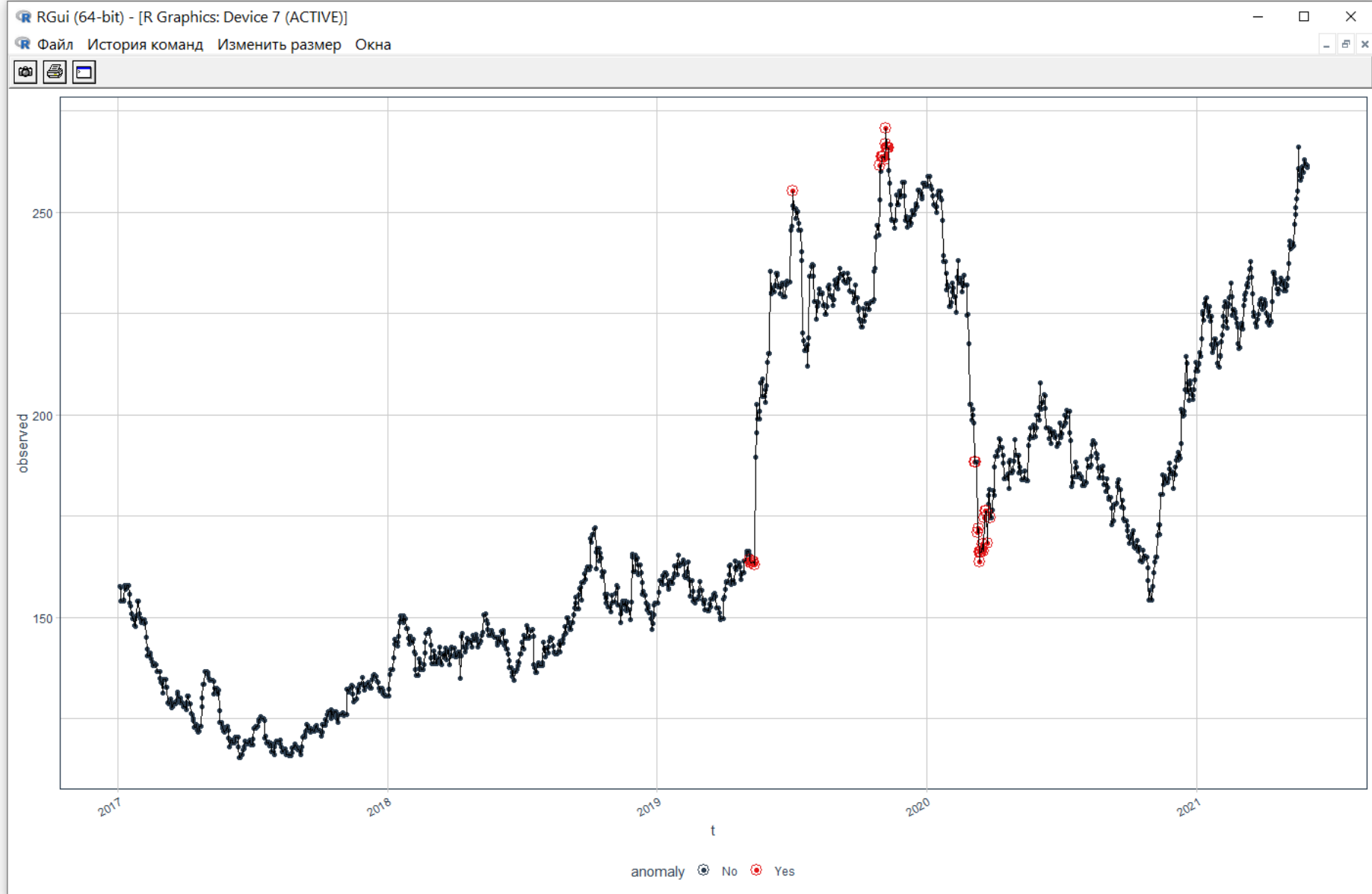
ДИНАМИКА ЦЕН АКЦИЙ ГАЗПРОМА С АВГУСТА 2020 ПО МАЙ 2021 И МНОВАРИАНТНЫЙ ПРОГНОЗ С 1 ИЮНЯ 2021 НА 25 РАБОЧИХ ДНЕЙ

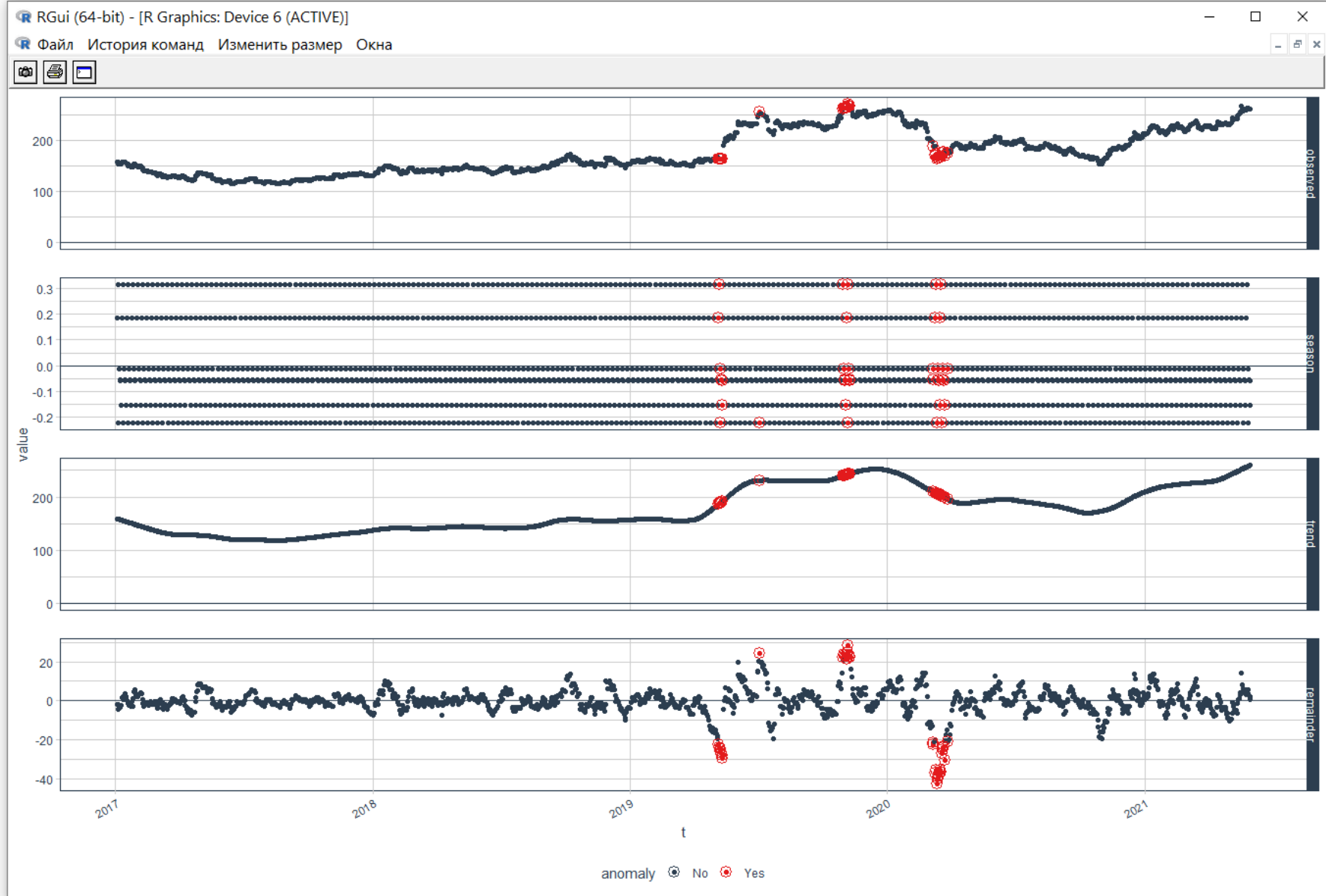












Row.names	Характеристика	Значение
1	trend_strength	0.998
2	seasonal_strength_week	0.185
3	seasonal_peak_week	3
4	seasonal_trough_week	6
5	spikiness	2.73e-05
6	linearity	1320
7	curvature	-149
8	stl_e_acf1	0.337
9	stl_e_acf10	0.241
10	acf1	0.997
11	acf10	9.63
12	diff1_acf1	0.0528
13	diff1_acf10	0.0164
14	diff2_acf1	-0.493
15	diff2_acf10	0.251
16	season_acf1	0.976
17	pacf5	0.997
18	diff1_pacf5	0.0104
19	diff2_pacf5	0.508
20	season_pacf	-0.00597
21	zero_run_mean	0
22	nonzero_squared_cv	0.0584
23	zero_start_prop	0
24	zero_end_prop	0
25	lambda_guerrero	-0.927
26	kpss_stat	12
27	kpss_pvalue	0.01
28	pp_stat	-0.586
29	pp_pvalue	0.1
30	ndiffs	1
31	nsdiffs	0
32	bp_stat	1600
33	bp_pvalue	0
34	lb_stat	1603
35	lb_pvalue	0
36	var_tiled_var	6.73e-05
37	var_tiled_mean	1
38	shift_level_max	34.3
39	shift_level_index	862
40	shift_var_max	331
41	shift_var_index	859
42	shift_kl_max	4.51
43	shift_kl_index	861
44	spectral_entropy	0.375
45	n_crossing_points	23
46	n_flat_spots	173
47	coef_hurst	1
48	stat_arch_lm	1

В последние годы большое развитие получили методы глубокого обучения (DL). Это связано с революционным прорывом в области аппаратного обеспечения (мощные графические процессоры, обеспечивающие эффективное распараллеливание вычислений), появлением возможности использовать, благодаря ИНТЕРНЕТУ, огромные массивы для обучения и тестирования нейронных сетей (NN) и разработкой эффективных алгоритмов и программного обеспечения (DL MatLab, tensorflow, keras) для обучения и тестирования сложнейших искусственных нейронных сетей различной архитектуры.

Естественно этот процесс затронул и проблемы, связанные с прогнозированием ВР, для этого стали широко применяться рекуррентные нейронные сети (RNN). Особенно успешно и широко используются в прогнозировании ВР так называемые сети с долговременной краткосрочной памятью (LSTM).

Для реализации технологии DL LSTM можно использовать различное программное обеспечение: tensorflow+keras, DL LSTM MatLab.

Приведем пример использования LSTM MatLab для прогноза ВР акций Газпрома, который мы уже рассмотрели выше.

Скрипт, реализующий технологию получения прогноза на базе построения и обучения NN LSTM:

```
%Предсказать данные временных рядов с помощью сети долгой краткосрочной памяти(LSTM).
%Прогнозирование временного ряда акций Газпром с использованием глубокого обучения
%Загружаем данные о ВР акций Газпрома
clc; clear all;
load Gazprom21_06

График ВР
figure
plot(Gazp202105)
xlabel("Day")
ylabel("Price")
title("Promotions Gazprom 2017,1,1-2021,5,31")

%Стандартизируем данные
mu = mean(Gazp202105);
sig = std(Gazp202105);
dataStandardized = (Gazp202105 - mu) / sig;

%Задаем интервалы обучения и теста
Ln0 = datenum('2017,1,1');%Дата начала ВР Gazp
L0 = datenum('2021,2,1');%Дата начала обучения модели
L = datenum('2021,5,31');%Дата окончания обучения модели
L0train=L0-Ln0;
Ltrain=L-Ln0;
Train_dan = dataStandardized(L0train:Ltrain);
```

%Подготавливаем предикторы и ответы

```
X = Train_dan(1:end-1);
```

```
Y = Train_dan(2:end);
```

%Задаем сетевую архитектуру LSTM

%Создаем сеть регрессии LSTM. Задаём слой LSTM, чтобы иметь 100 скрытых модулей.

```
numFeatures = 1;
```

```
numResponses = 1;
```

```
numHiddenUnits = 100;
```

```
layers = [ ...
```

```
    sequenceInputLayer(numFeatures)
```

```
    lstmLayer(numHiddenUnits)
```

```
    fullyConnectedLayer(numResponses)
```

```
    regressionLayer];
```

%Задаём опции обучения. Установить решатель на 'adam' и обучать в течение 250 эпох.

%Чтобы препятствовать тому, чтобы градиенты взорвались, установить порог градиента 1.

%Указать начальную скорость обучения InitialLearnRate = 0.005 и установить

%период снижения скорости обучения LearnRateDropPeriod после 150 эпох

%путем умножения на фактор LearnRateDropFactor = 0,15.

```
options = trainingOptions('adam', ...
```

```
    'MaxEpochs',250, ...
```

```
    'GradientThreshold',1, ...
```

```
    'InitialLearnRate',0.005, ...
```

```
    'LearnRateSchedule','piecewise', ...
```

```
    'LearnRateDropPeriod',150, ...
```

```
    'LearnRateDropFactor',0.15, ...
```

```
    'Verbose',0, ...
```

```
    'Plots','training-progress');
```

%Обучаем сеть LSTM с заданными опциями обучения при помощи trainNetwork.

```
net = trainNetwork(X,Y,layers,options);
```

%предсказание BP на обучающей выборке

```
net = predictAndUpdateState(net,X);
```

```
[net,YPred] = predictAndUpdateState(net,Y(1:end));
```

%Переходим от стандартизируемых предсказаний к исходному уровню

%с помощью параметров, вычисленных ранее.

```
Y0Pred = sig*YPred + mu;
```

%График процесса обучения сообщает о среднеквадратичной ошибке (RMSE), вычисленной

%из стандартизированных данных. Вычислим RMSE из нестандартизированных предсказаний.

```
Yreal = Gazp202105(L0train+1:Ltrain);
```

```
rmse = sqrt(mean((Y0Pred-Yreal).^2));
```

%Построим временные ряды с предсказанными значениями.

%Сравним предсказанные значения с фактическими данными.

```
%----- train -----
```

```
figure
```

```
subplot(2,1,1)
```

```
plot(Yreal)
```

```
hold on
```

```
plot(Y0Pred,'-')
```

```

hold off
xlabel("Day")
ylabel("Price")
title("Forecast")
legend(["Observed" "Forecast"])
subplot(2,1,2)
stem(Y0Pred - Yreal)
xlabel("Day")
ylabel("Error")
title("RMSE = " + rmse)
if Ltrain < numel(Gazp202105)

%расчет данных на тесте
%Xtest = Gazp202105(Ltrain+1:end-1);
h=numel(Gazp202105)-Ltrain;
h0=numel(Y);
hk =h0+h-1;

for i = h0:hk
    [net,YPred(:,i)] = predictAndUpdateState(net,YPred(:,i-1),'ExecutionEnvironment','cpu');
end

%[net,YPred_test] = predictAndUpdateState(net2,Xtest);

Y0Pred_test = sig*YPred(h0+1:hk)+ mu;
Yreal_test = Gazp202105(Ltrain+2:end);
rmse_t = sqrt(mean((Y0Pred_test-Yreal_test).^2));

%----- test -----
figure
subplot(2,1,1)
plot(Yreal_test)
hold on
plot(Y0Pred_test,'-')
hold off
xlabel("Day")
ylabel("Price")
title("Forecast")
legend(["Observed" "Forecast"])

subplot(2,1,2)
stem(Y0Pred_test - Yreal_test)
xlabel("Day")
ylabel("Error")
title("RMSE test = " + rmse_t)

else

%Расчет прогнозных значений и их графическое представление
%Задаем горизонт прогнозирования в рабочих днях
h=25;
h0=numel(Y);
hk =h0+h-1;

for i = h0:hk
    [net,YPred(:,i)] = predictAndUpdateState(net,YPred(:,i-1),'ExecutionEnvironment','cpu');
end

```

```

%Переходим от стандартизируемых предсказаний к исходному уровню
YPrognoz = sig*YPred + mu;
%Выдаем массив прогнозов
Yh=YPrognoz(h0:hk);
fprintf('%0.2f\n',Yh);

%Строим график ВР + ПРОГНОЗ
figure
plot(Yreal)
hold on
plot(Y0Pred,'-g','LineWidth',2)
hold on
idx = h0:hk;
plot(idx,YPrognoz(h0:hk),'-','LineWidth',3)
hold off
xlabel("Day")
ylabel("Price")
title("Forecast")
legend(["Observed" "Forecast train" "Прогноз"])

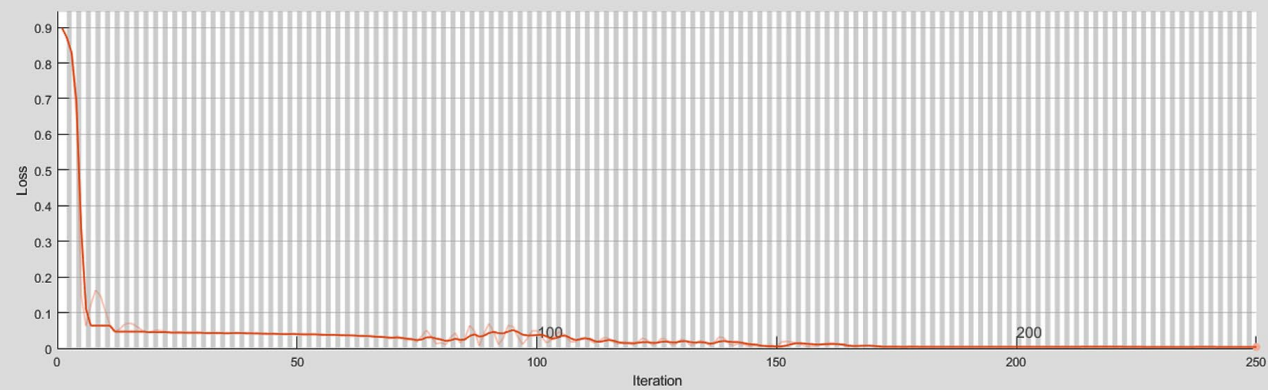
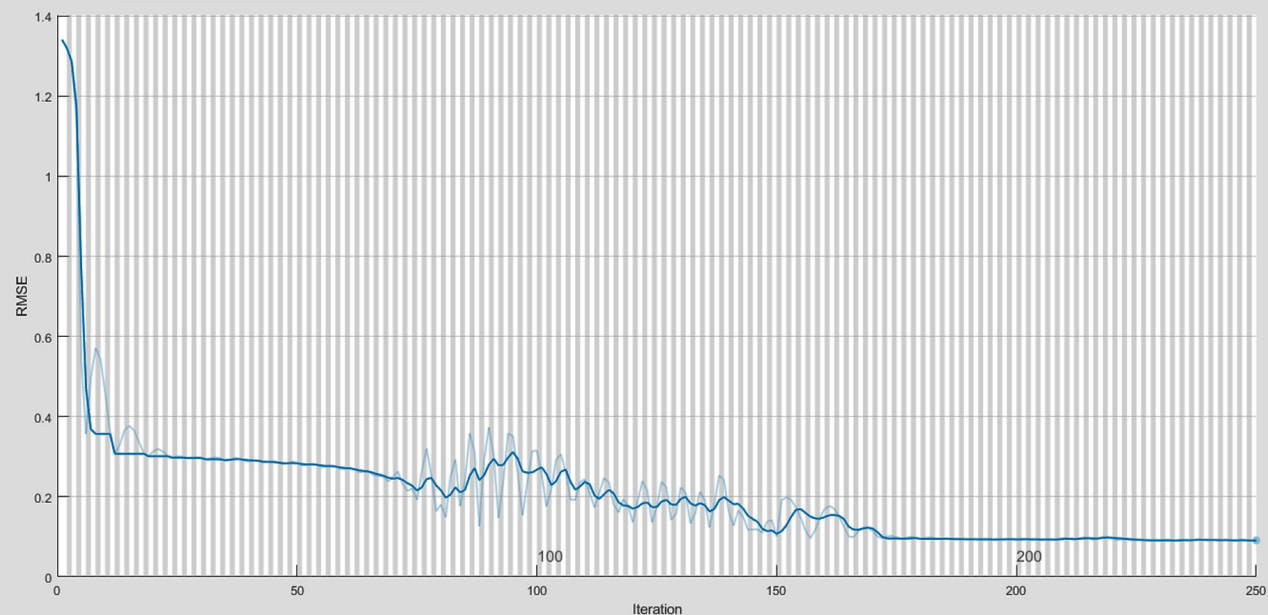
end

```

Скрипт позволяет выделять любой кусок ВР для обучения LSTM, оставшаяся часть ряда используется для теста. При прогнозировании интервал обучения заканчивается последней фактической точкой ВР и прогноз осуществляется последовательно шаг за шагом пока не будет достигнут конец заданного горизонта прогнозирования. Но с учетом того, что LSTM хранит в памяти определенную предысторию значение в точке $t+1$ определяется некоторым набором значений в предыдущие моменты времени. Глубина памяти LSTM задается ее архитектурой и рядом параметров.

Графические и числовые результаты работы скрипта представлены ниже:

2021,1,1 - 2021,5,31

**Results**

Validation RMSE: N/A
Training finished: Reached final iteration

Training Time

Start time: 04-Jun-2021 19:22:57
Elapsed time: 15 sec

Training Cycle

Epoch: 250 of 250
Iteration: 250 of 250
Iterations per epoch: 1
Maximum iterations: 250

Validation

Frequency: N/A
Patience: N/A

Other Information

Hardware resource: Single GPU
Learning rate schedule: Piecewise
Learning rate: 0.00075

[Learn more](#)**RMSE**

— Training (smoothed)
—•— Training
- - • - - Validation

Loss

— Training (smoothed)
—•— Training
- - • - - Validation

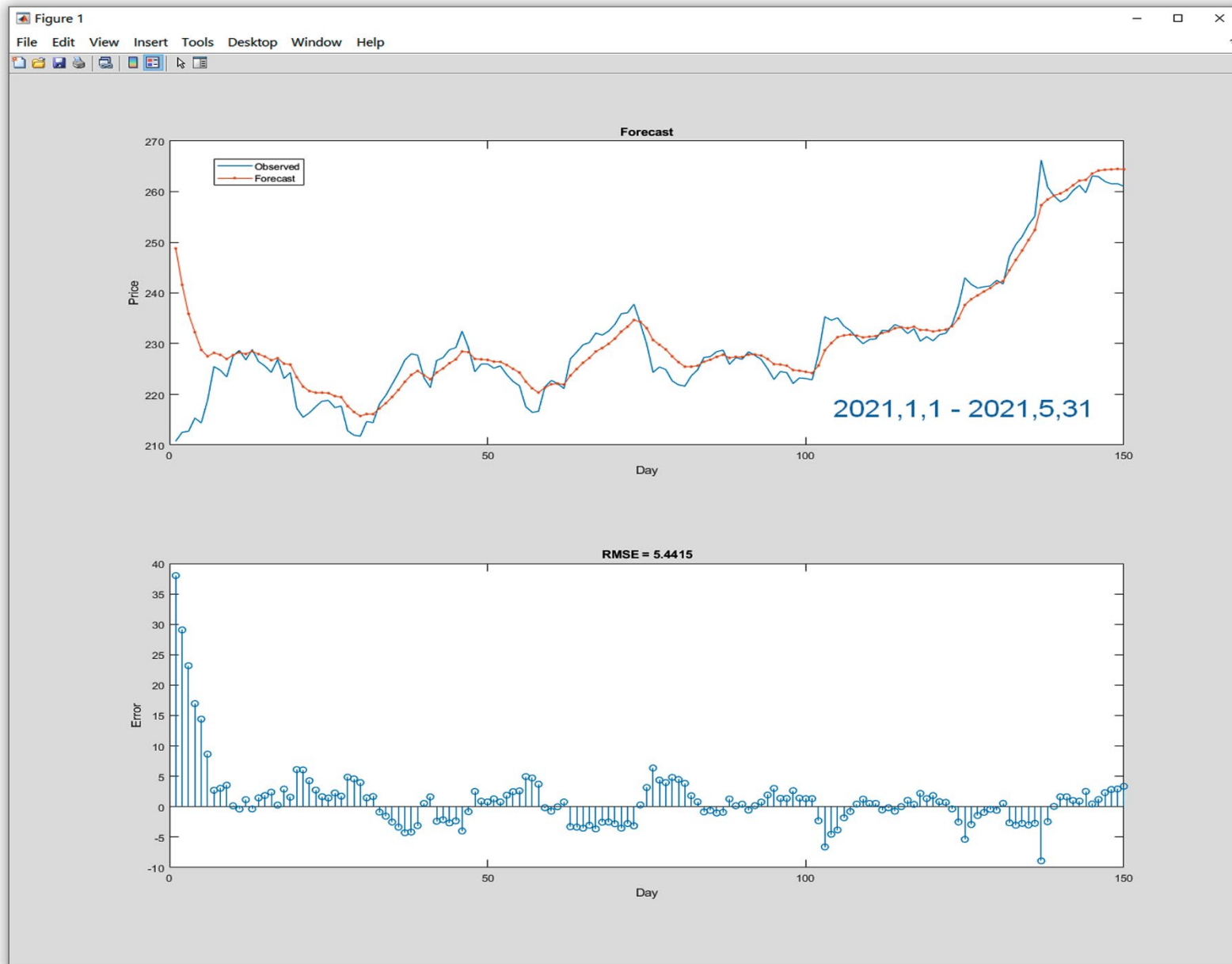


Figure 2

File Edit View Insert Tools Desktop Window Help



265.42
266.16
266.83
267.44
268.00
268.51
268.98
269.41
269.79
270.15
270.46
270.75
271.00
271.23
271.44
271.62
271.78
271.93
272.06
272.17
272.27
272.36
272.44
272.51
272.57

