

Прогноз акций Газпром на май 2021 года DL NN Tensorflow Keras

Код python

```
'''python
import numpy as np
from pandas import Series, DataFrame
import pandas as pd
import matplotlib.pyplot as plt
import openpyxl
import xlrd
import math
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
import tensorflow as tf
from tensorflow.keras import layers
from tensorflow.keras.layers.experimental import preprocessing

np.random.seed(50)

#ввод временного ряда и его отображение
data_ts =pd.ExcelFile('D:\Machine learning\DATA\AkzGASP.xlsx')
ts_dat =data_ts.parse('GAZP')

plt.figure(1, figsize=(12,8))
plt.plot(ts_dat)
plt.show()
#print(ts_dat)

#формирование вектор столбца из значений временного ряда
dataset1 = np.array(ts_dat)
dataset1 = dataset1.astype('float32')
#print(dataset1)

#нормирование значений временного ряда
scaler = MinMaxScaler(feature_range=(0, 1))
dataset = scaler.fit_transform(dataset1)
```

```

#print('нормированный ts')
#print(dataset)

# получение train и test выборок из ts
train_size = int(len(dataset) * 0.9)
test_size = len(dataset) - train_size
train, test = dataset[0:train_size,:], dataset[train_size:len(dataset),:]
#print('train ts')
#print(train)

#функция для формирования trainX, trainY и testX, testY
def create_dataset(dataset, look_back= 1):
    dataX, dataY = [], []
    for i in range(len(dataset)-look_back-1):
        a = dataset[i:(i+look_back), 0]
        dataX.append(a)
        dataY.append(dataset[i + look_back, 0])
    return np.array(dataX), np.array(dataY)

look_back = 1
trainX, trainY = create_dataset(train, look_back)
testX, testY = create_dataset(test, look_back)

# КОНСТРУИРОВАНИЕ, КОМПИЛИРОВАНИЕ И ОБУЧЕНИЕ NN

ts_model = tf.keras.Sequential([
    layers.Dense(64),
    layers.Dense(18),
    layers.Dense(1)
])

ts_model.compile(loss = tf.losses.MeanSquaredError(),
                  optimizer = tf.optimizers.Adam())
ts_model.fit(trainX, trainY, epochs=20)
scores = ts_model.evaluate(testX, testY, verbose=0)
#print(scores)

```

```

# ПОЛУЧЕНИЕ ПРОГНОЗОВ НА TRAIN, TEST
# И ПЕРЕВОД ИХ В ИСХОДНЫЙ МАСШТАБ

pred_train=ts_model.predict(trainX)
pred_test=ts_model.predict(testX)
pred_train = scaler.inverse_transform(pred_train)
trainY = scaler.inverse_transform([trainY])
pred_test = scaler.inverse_transform(pred_test)
testY = scaler.inverse_transform([testY])
trainScore = math.sqrt(mean_squared_error(trainY[0], pred_train[:,0]))
print('Train Score: %.2f RMSE' % (trainScore))
testScore = math.sqrt(mean_squared_error(testY[0], pred_test[:,0]))
print('Test Score: %.2f RMSE' % (testScore))

#plt.plot(ts_dat[1035:2890])
#plt.plot(pred_test[1934:2890,:])
#plt.plot(pred_train)
#plt.show()

#формирование прогноза

h=22
prognoz=np.ones((h, 1), dtype=float)
prognoz[0]=testX[len(testX)-1]

for i in range(h-1):
    prognoz[i+1] = ts_model.predict(prognoz[i])
    #print(prognoz[i+1])
prognoz=scaler.inverse_transform(prognoz)
print(prognoz)

#графическое отображение результатов

```

```

ts= dataset1
test_ts= pred_test
train_ts= pred_train
prognoz_ts= prognoz

```

```

Prognoz =DataFrame(prognoz_ts, columns=['GAZP'],
    index=range(len(ts), len(ts)+22))
ts =DataFrame(ts, columns=['GAZP'])
test =DataFrame(test_ts, columns=['GAZP'],
    index=range(len(train_ts), len(train_ts)+len(test_ts)))

```

```

plt.figure(2, figsize=(12,8))
plt.plot(ts, color='blue', linewidth=0.2,
    label = 'Исходный временной ряд')
plt.plot(train_ts, color='violet', linewidth=1.2,
    label = 'Результаты обучения')
plt.plot(test, color='green', linewidth=0.8,
    label = 'Результаты теста ')
plt.plot(Prognoz, color='red', linewidth=3 ,
    label = 'Прогноз на 22 дня сначала мая')
plt.title("Прогноз акций GAZP DL NN tf.keras")
plt.xlabel('time')
plt.ylabel('Цена закрытия, руб.')
plt.legend(loc="best")
plt.grid(True)
plt.show()

```

```

#сохранение данных

```

```
#np.save('D:/Machine learning/PY_tf/ts', dataset1)
#np.save('D:/Machine learning/PY_tf/ts_train', pred_train)
#np.save('D:/Machine learning/PY_tf/ts_test', pred_test)
#np.save('D:/Machine learning/PY_tf/ts_prognoz2', prognoz)

'''
```

Результаты моделирования ts GAZP:

```
Epoch 1/20
82/82 [=====] - 0s 481us/step - loss: 0.0042

Epoch 2/20
82/82 [=====] - 0s 456us/step - loss: 2.9099e-04

Epoch 3/20
82/82 [=====] - 0s 476us/step - loss: 3.0042e-04

Epoch 4/20
82/82 [=====] - 0s 456us/step - loss: 2.9745e-04

Epoch 5/20
82/82 [=====] - 0s 456us/step - loss: 2.9280e-04

Epoch 6/20
82/82 [=====] - 0s 456us/step - loss: 3.1482e-04

Epoch 7/20
82/82 [=====] - 0s 456us/step - loss: 2.9741e-04

Epoch 8/20
82/82 [=====] - 0s 473us/step - loss: 2.9839e-04

Epoch 9/20
82/82 [=====] - 0s 456us/step - loss: 3.0549e-04
```

```

Epoch 10/20
82/82 [=====] - 0s 468us/step - loss: 2.9083e-04

Epoch 11/20
82/82 [=====] - 0s 456us/step - loss: 2.9726e-04

Epoch 12/20
82/82 [=====] - 0s 468us/step - loss: 3.1711e-04

Epoch 13/20
82/82 [=====] - 0s 491us/step - loss: 2.9850e-04

Epoch 14/20
82/82 [=====] - 0s 443us/step - loss: 2.9854e-04

Epoch 15/20
82/82 [=====] - 0s 443us/step - loss: 3.2412e-04

Epoch 16/20
82/82 [=====] - 0s 466us/step - loss: 3.3617e-04

Epoch 17/20
82/82 [=====] - 0s 443us/step - loss: 2.9850e-04

Epoch 18/20
82/82 [=====] - 0s 457us/step - loss: 3.1358e-04

Epoch 19/20
82/82 [=====] - 0s 497us/step - loss: 3.3132e-04

Epoch 20/20
82/82 [=====] - 0s 468us/step - loss: 3.0412e-04

Train Score: 2.93 RMSE

Test Score: 2.92 RMSE

```

Прогноз Май 2021 на 22 дня цена закрытия акций GAZP:

[[232.95999903]
[234.39956417]
[235.84955749]
[237.31007644]
[238.7811795]
[240.26296412]
[241.75546929]
[243.25881196]
[244.7730701]
[246.29831193]
[247.83461542]
[249.38205854]
[250.94074848]
[252.5107145]
[254.09208328]
[255.6849133]
[257.28930203]
[258.90532742]
[260.53305771]
[262.17259034]
[263.82404228]
[265.48746225]]

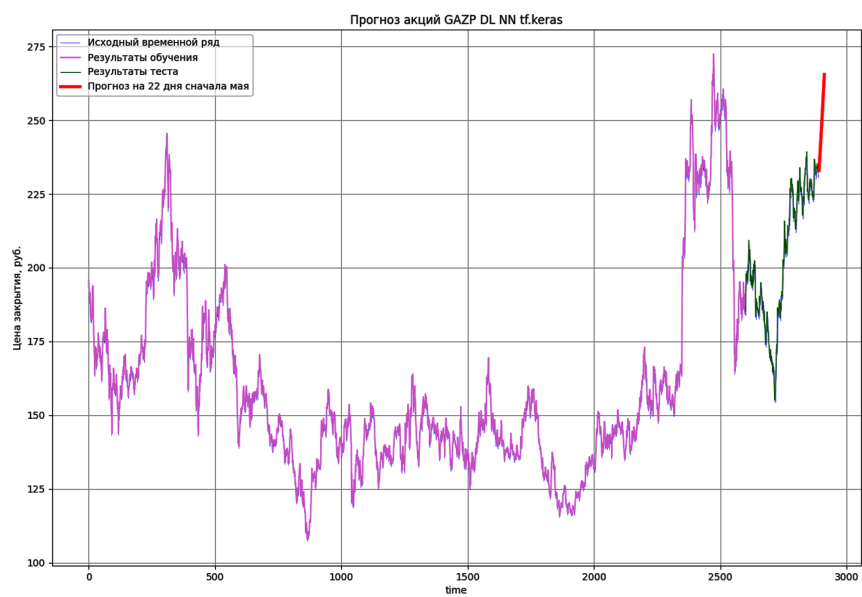


Рис. 1: Прогноз Май 2021 на 22 дня цена закрытия акций GAZP