

Технический анализ акции РусГидро на Python


Валова Екатерина Владимировна

Оглавление



Данные	2
Код на Python.....	3
График 1. Полосы Боллинджера + Аллигатор Билла Уильямса.....	6
График 2. Стохастик + Индекс относительной силы (RSI)	8
Выводы. Полосы Боллинджера	10
Аллигатор Билла Уильямса	11
Стохастик	12
Индекс относительной силы (RSI)	13
Итог: покупать	14

Данные


Экспорт в MetaStock // MFD URL: <https://mfd.ru/> (дата обращения: 28.11.2023).


Таймфрейм День 



Интервал 28.11.2023 – 28.11.2023

Формат Текстовый  Все тикеры в одном файле 


Имя файла РусГидро_1day_28112023_28112023.txt

Разделитель полей Точка с запятой 

Десятичный разделитель Точка 

Формат даты / времени dd/MM/yy  HHmmss 

Добавить заголовок файла ☒

Формат записи TICKER,PER,DATE,TIME,OPEN,HIGH,LOW,CLOSE,VOL,OPENINT 

Заполнять периоды без сделок ☒

Код на Python

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

'''
                                ТЕХНИЧЕСКИЙ АНАЛИЗ АКЦИЙ
'''

#подготовка файла
Stock_of_RG = pd.read_csv('C:\RusGidro.txt', sep=';')
Stock_of_RG = Stock_of_RG.rename(columns = {'<DATE>':'date', '<CLOSE>':'close',
'<OPEN>':'open',
                                '<HIGH>':'high', '<LOW>':'low',
'<VOL>':'vol'})
Stock_of_RG.date = pd.to_datetime(Stock_of_RG.date, format='%d/%m/%y')
Stock_of_RG = Stock_of_RG.drop(columns = ['<PER>', '<TICKER>', '<TIME>',
'<OPENINT>'], axis = 1)

#японские свечи
plt.figure(1)
#ширина свечей
width = .6
width2 = .1
#определяем, где цена падает, а где возрастает
up = Stock_of_RG[Stock_of_RG.close >= Stock_of_RG.open ]
down = Stock_of_RG[Stock_of_RG.close < Stock_of_RG.open ]
#цвета для возрастающей (зеленый) и убывающей (красный)
col1 = 'palegreen'
col2 = 'lightcoral'

#зеленые свечи
plt.bar(up.date, up.close -up.open ,width, bottom=up.open, color=col1)
plt.bar(up.date,up.high -up.close ,width2, bottom=up.close, color=col1)
plt.bar(up.date,up.low -up.open ,width2, bottom=up.open, color=col1)
#красные свечи
plt.bar(down.date, down.close -down.open, width, bottom=down.open, color=col2)
plt.bar(down.date, down.high -down.open, width2, bottom=down.open, color=col2)
plt.bar(down.date, down.low -down.close, width2, bottom=down.close, color=col2)
'''
                                Вычисление полос Боллинджера
'''

#скользящая средняя с шагом step_N
step_N = 10
len_stock = len(Stock_of_RG.close)
#скользящая средняя
moving_average = [sum(Stock_of_RG.close[i-step_N:i])/step_N for i in range(step_N,
len_stock +1)]
#скользящее отклонение
moving_std = [np.std(Stock_of_RG.close[i-step_N:i]) for i in range(step_N,
len_stock +1)]
#полосы Боллинджера
lower_bollinger = [(moving_average[i-step_N] - 2*moving_std[i-step_N]) for i in
range(step_N, len_stock +1)]
upper_bollinger = [moving_average[i-step_N] + 2*moving_std[i-step_N] for i in
range(step_N, len_stock +1)]
#ниже закомментирована скользящая средняя, на графике она не нужна
#plt.plot(Stock_of_RG.date[step_N-1 : len_stock], moving_average, label='moving
average', linewidth=1.0)
#рисует полосы
plt.plot(Stock_of_RG.date[step_N-1 : len_stock], lower_bollinger, linewidth=.3,
color='purple')
plt.plot(Stock_of_RG.date[step_N-1 : len_stock], upper_bollinger, label='line
```

```

bollinger', linewidth=.3, color='purple')
#закрашиваем между полос
plt.fill_between(Stock_of_RG.date[step_N-1 : len_stock],
                 upper_bollinger, lower_bollinger, color='purple', alpha=0.1,
                 label='Bollinger Bands')

#подготовка значений для оси X
first_date = Stock_of_RG.date[0]
last_date = Stock_of_RG.date[len_stock-1]
our_dates = pd.date_range(start = first_date, periods = len_stock, freq='D')
new_dates = pd.date_range(start=last_date + pd.Timedelta(days=1), periods = 8,
                           freq='D')
date_for_alligator = our_dates.tolist() + new_dates.tolist() #значения X для
Аллигатора Билла Уильямса
'''

Аллигатор Билла Уильямса

Челюсть Аллигатора, или синяя линия, - это 13-периодная
сглаженная скользящая средняя,
смещенная вперед на 8 баров
'''
period_alligatora = 13 #период скользящей средней
sdvig_alligatora = 8 #сдвиг скользящей средней
chealust_alligatora = [sum(Stock_of_RG.close[i-
period_alligatora:i])/period_alligatora
                        for i in range(period_alligatora, len_stock +1)] #значение
Y
len_alligatora = len(chealust_alligatora) #длина скользящей средней
plt.plot(date_for_alligator[period_alligatora+sdvig_alligatora -1 :
period_alligatora+len_alligatora+sdvig_alligatora],
         chealust_alligatora, label='chealust alligatora', linewidth=1.0,
         color='deepskyblue') #Челюсть Аллигатора
'''

Зубы Аллигатора, или красная линия, - это 8-периодная
сглаженная скользящая средняя,
смещенная вперед на 5 баров
'''
period_alligatora = 8 #период скользящей средней
sdvig_alligatora = 5 #сдвиг скользящей средней
zub_alligatora = [sum(Stock_of_RG.close[i-period_alligatora:i])/period_alligatora
                  for i in range(period_alligatora, len_stock +1)] #значение Y
len_alligatora = len(zub_alligatora) #длина скользящей средней
plt.plot(date_for_alligator[period_alligatora+sdvig_alligatora :
period_alligatora+len_alligatora+sdvig_alligatora],
         zub_alligatora, label='zub alligatora', linewidth=1.0,
         color='lightcoral') #Зубы Аллигатора
'''

Губы Аллигатора, или зеленая линия, - это 5-периодная сглаженная
скользящая средняя,
смещенная вперед на 3 бара
'''
period_alligatora = 5 #период скользящей средней
sdvig_alligatora = 3 #сдвиг скользящей средней
guba_alligatora = [sum(Stock_of_RG.close[i-period_alligatora:i])/period_alligatora
                   for i in range(period_alligatora, len_stock +1)] #значение Y
len_alligatora = len(guba_alligatora) #длина скользящей средней
plt.plot(date_for_alligator[period_alligatora+sdvig_alligatora :
period_alligatora+len_alligatora+sdvig_alligatora],
         guba_alligatora, label='guba alligatora', linewidth=1.0,
         color='palegreen') #Губы Аллигатора

#подписи на графике
plt.xticks (rotation= 0 , ha='right')
plt.xlabel('date', fontsize=15)

```

```

plt.ylabel('price', fontsize=15)
plt.title('РусГидро, HYDR', fontsize=17)
plt.legend()
'''
'''
Осциллятор — стохастик
'''
plt.figure(2)
#период для осциллятора K
K_period = 14
K_stochastic = [(100*(Stock_of_RG.close[i]-min(Stock_of_RG.low[i-K_period:i]))
                /(max(Stock_of_RG.high[i-K_period:i])-min(Stock_of_RG.low[i-K_period:i]))))
                for i in range (K_period, len_stock)]
plt.plot(Stock_of_RG.date[K_period:len_stock], K_stochastic, label='%K',
linewidth=1.0, color='deepskyblue')
#период для осциллятора D
D_period = 3
D_stochastic = [sum(K_stochastic[i-D_period:i])/D_period for i in
range(D_period, len(K_stochastic))]
plt.plot(Stock_of_RG.date[len_stock-len(D_stochastic):len_stock], D_stochastic,
label='%D',
linewidth=1.0, color='coral', linestyle=':')
'''
'''
Индекс относительной силы — RSI
(Relative Strength index)
'''
#период
period=14

#абсолютный прирост
growth_price = [Stock_of_RG.close[i] - Stock_of_RG.close[i - 1] for i in range(1,
len_stock)]

#промежуточные вычисления
RS = [0 for i in range(len_stock-period-1)]
for i in range (period, len_stock-1):
    sum_of_growth, sum_of_fall= 0, 0
    col_of_growth, col_of_fall =0, 0
    for j in range (i-period, i):
        if growth_price[j]>0:
            sum_of_growth += growth_price[j]
            col_of_growth += 1
        else:
            sum_of_fall -= growth_price[j]
            col_of_fall += 1
    RS[i-period] = (sum_of_growth / col_of_growth) / (sum_of_fall / col_of_fall)

#Relative Strength index
RSI=[100-100/(1+RS[i]) for i in range(len(RS))]

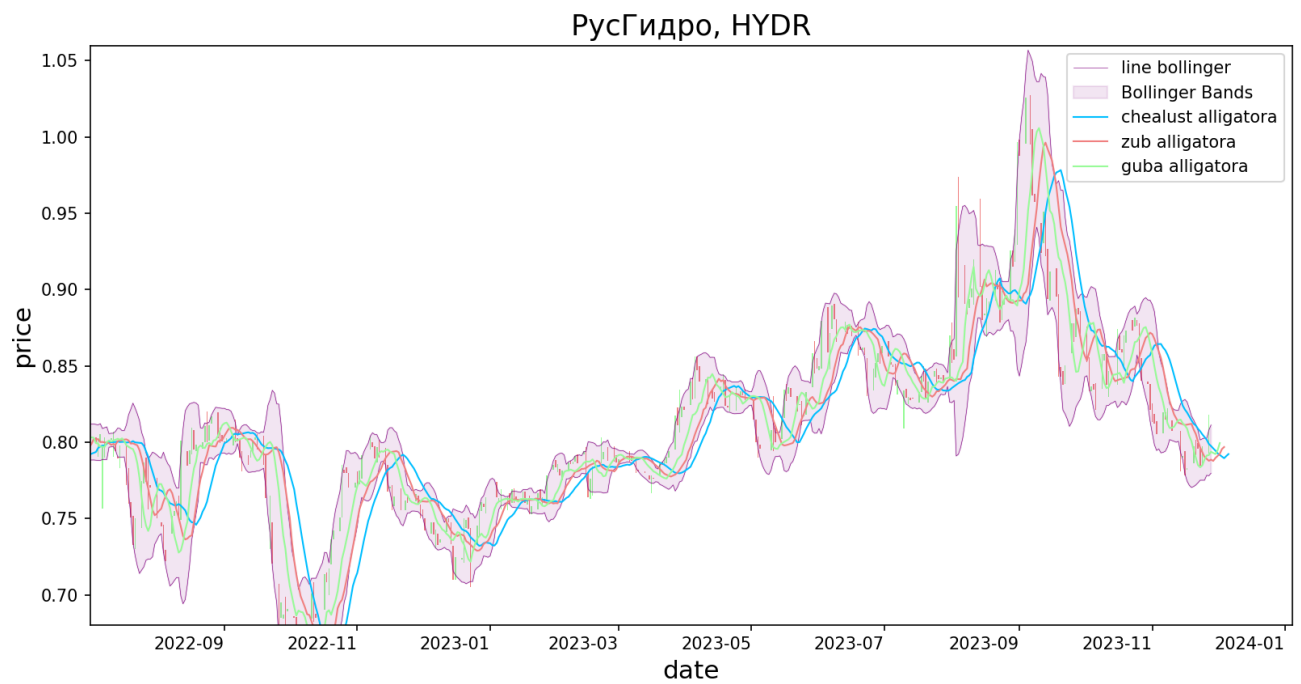
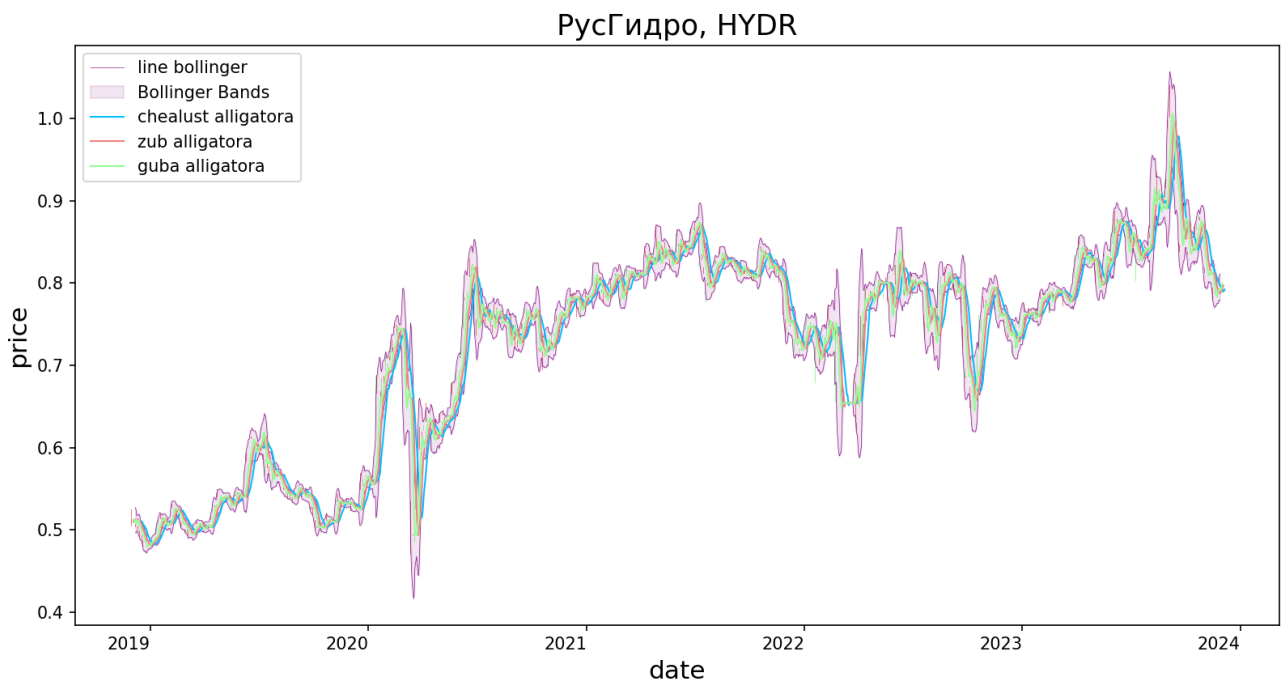
plt.plot(Stock_of_RG.date[period+1:], RSI[:len_stock], label='Relative Strength
index', linewidth=1.0, color='purple')
plt.plot(Stock_of_RG.date[period+1:], [30 for i in range(len_stock-period-1)],
label='signal boundary for PSI', linewidth=1.0, color='coral')
plt.plot(Stock_of_RG.date[period+1:], [70 for i in range(len_stock-period-1)],
linewidth=1.0, color='coral')

#форматируем график
plt.xlabel('date', fontsize=15)
plt.ylabel('percent, %', fontsize=15)
plt.title('РусГидро, HYDR', fontsize=17)
plt.legend()

#СМОТРИМ график
plt.show()

```

График 1. Полосы Боллинджера + Аллигатор Билла Уильямса



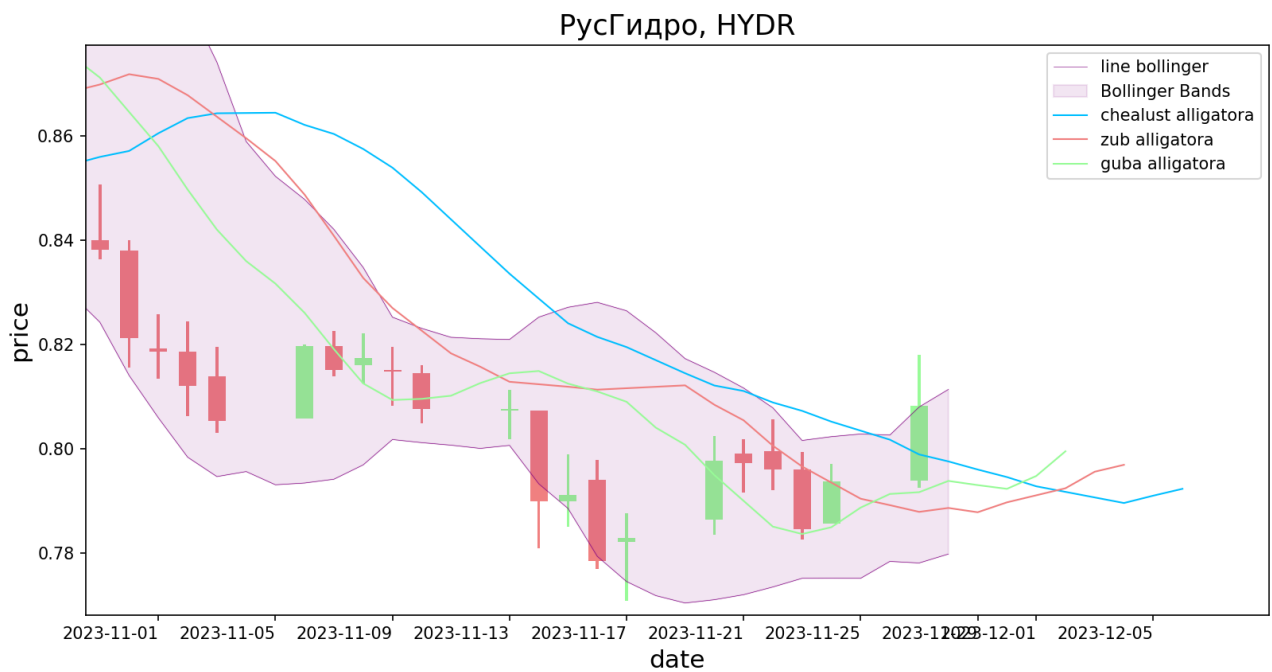
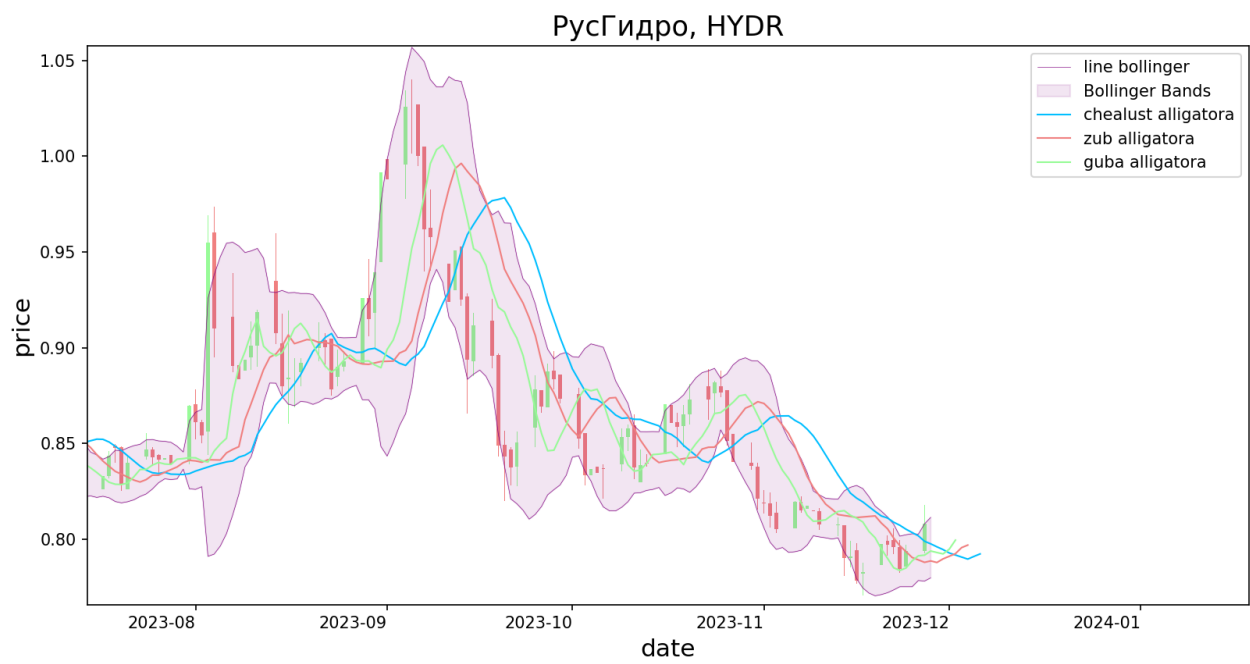
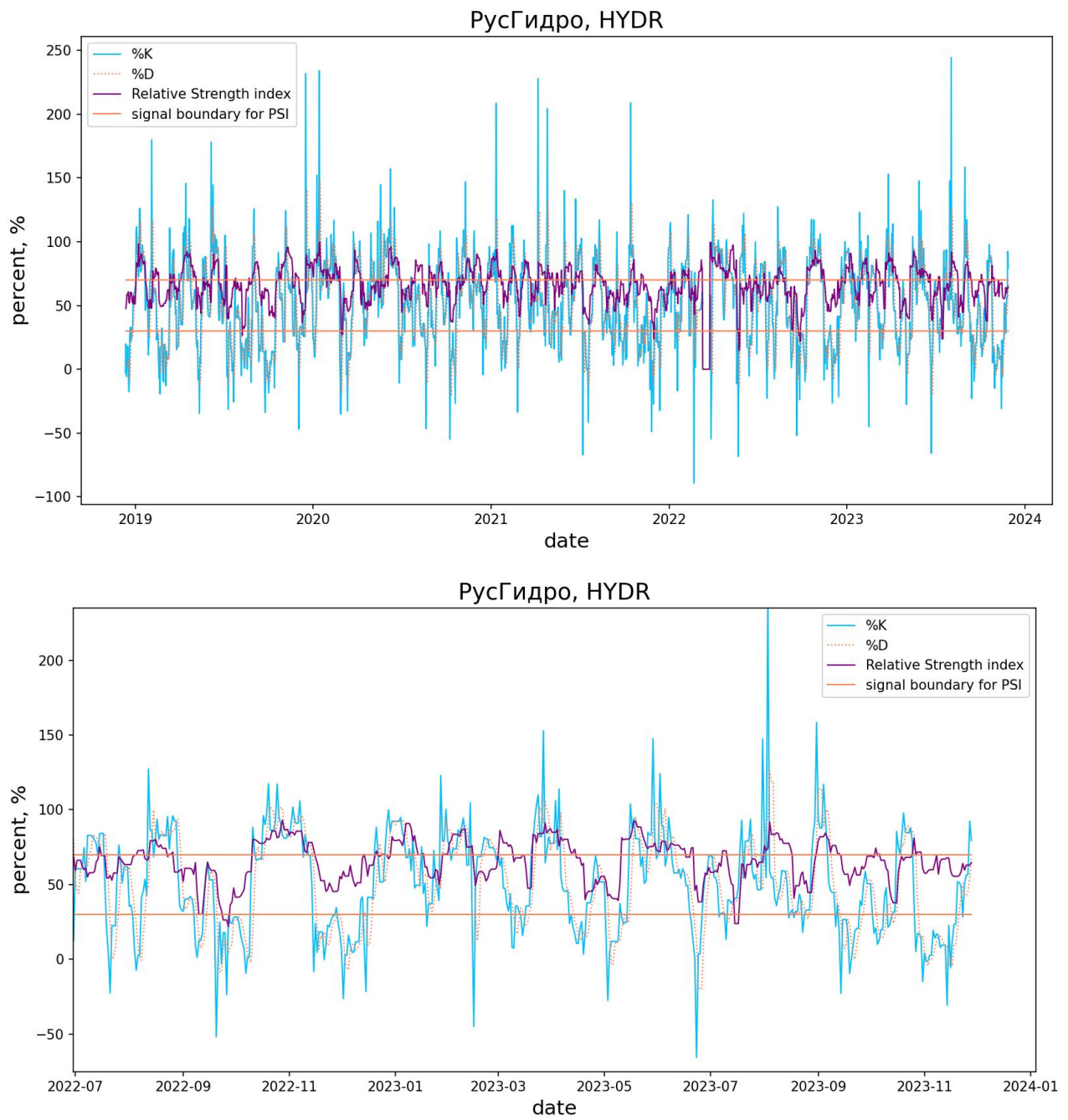
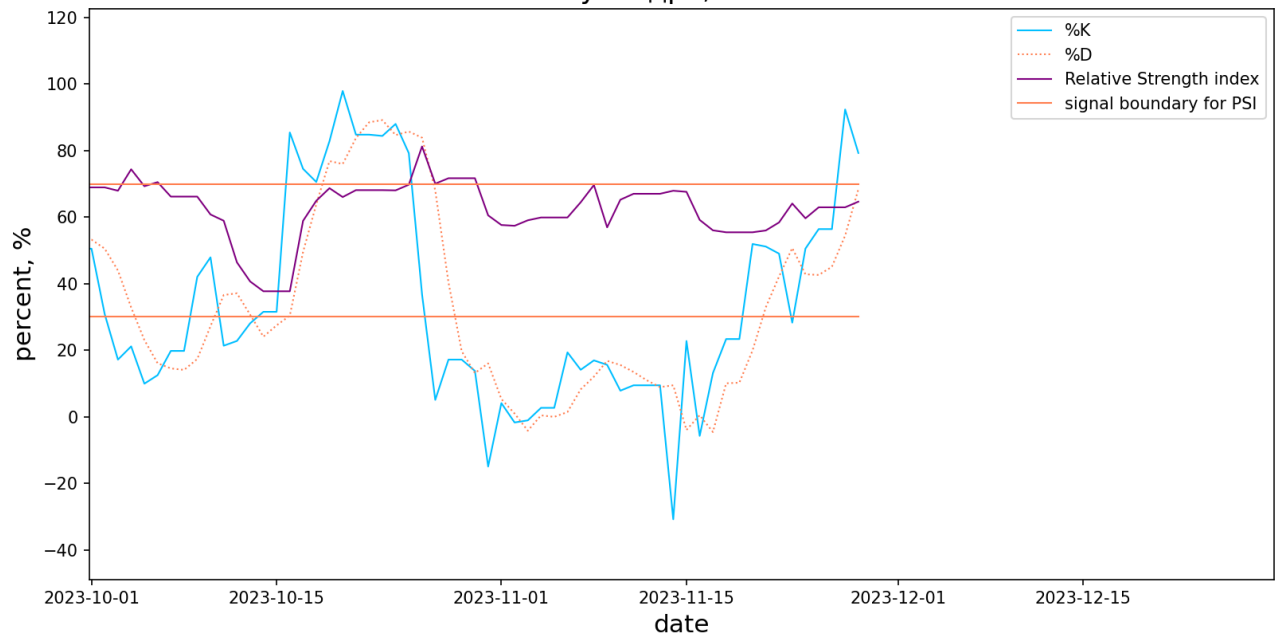


График 2. Стохастик + Индекс относительной силы (RSI)

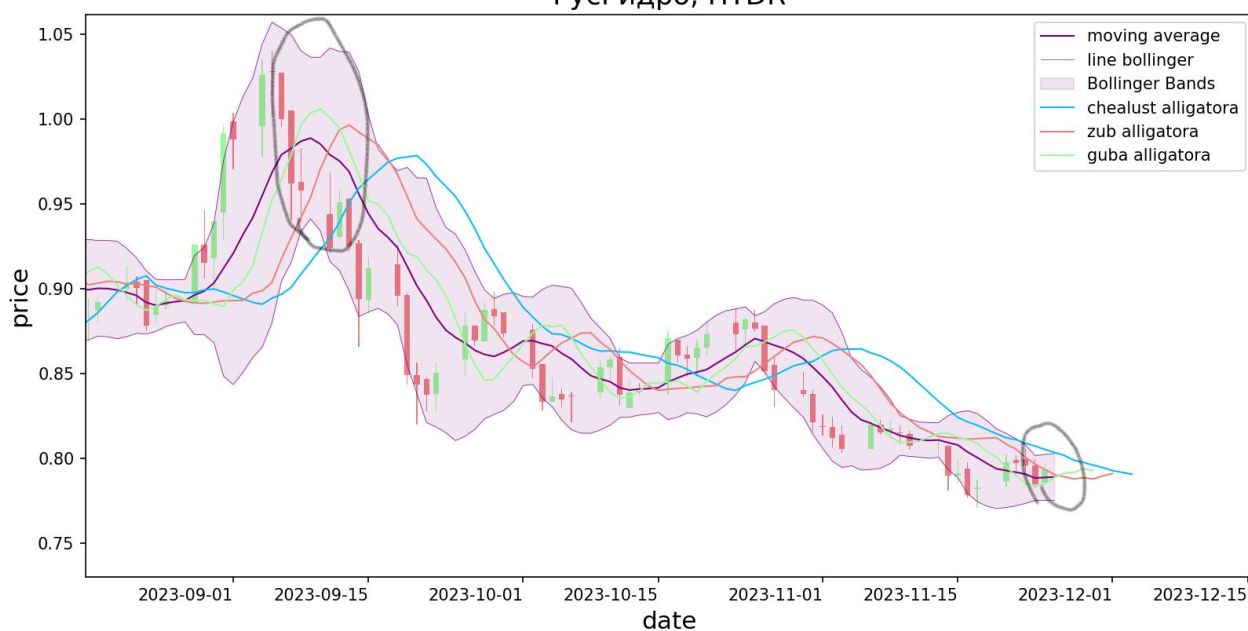


РусГидро, HYDR



Выводы. Полосы Боллинджера

РусГидро, HYDR



Внешние полосы Боллинджера сужаются, это может свидетельствовать о затухании тренда и возможном развороте.

РусГидро, HYDR

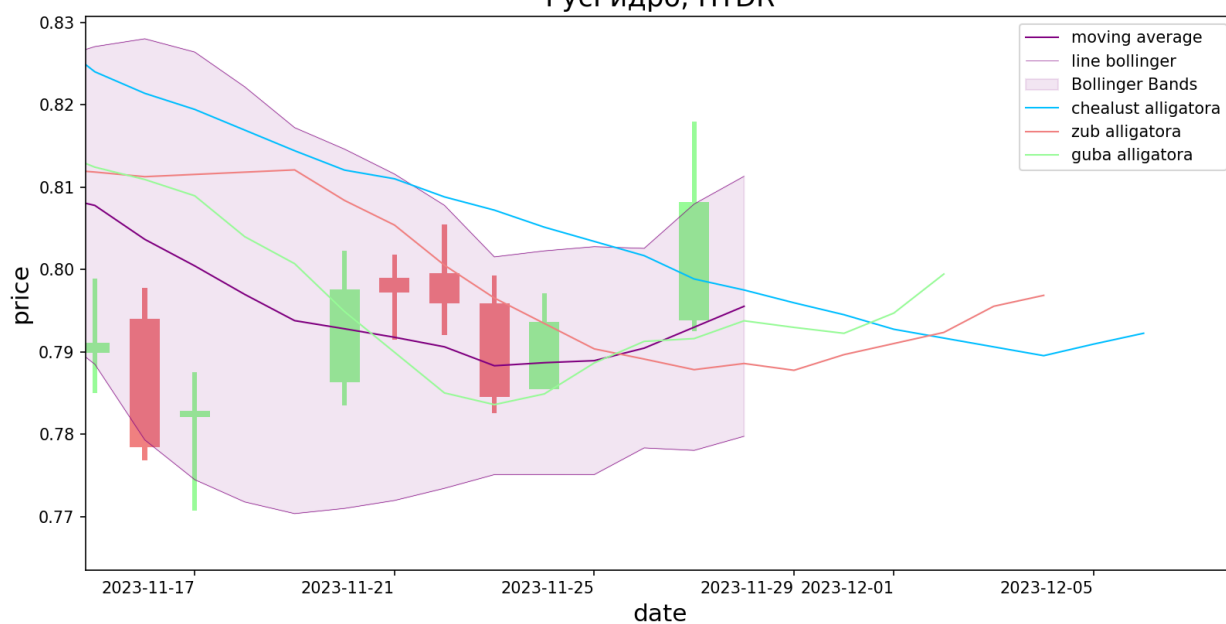
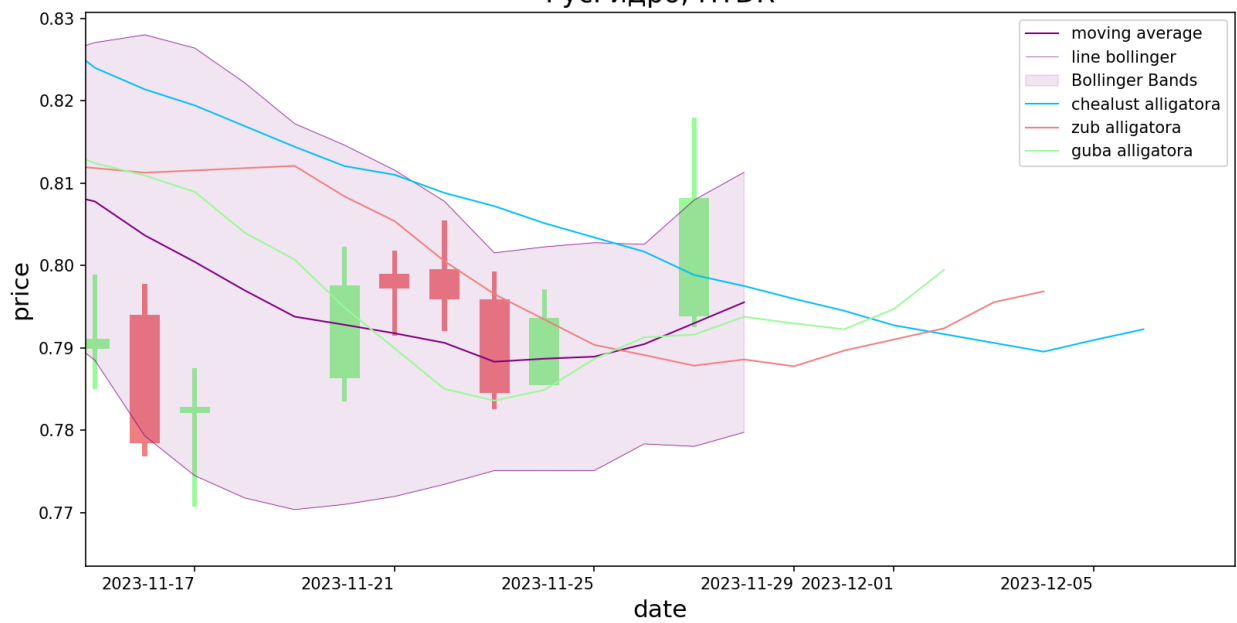


График выше средней, тренд восходящий.

Аллигатор Билла Уильямса

РусГидро, NYDR



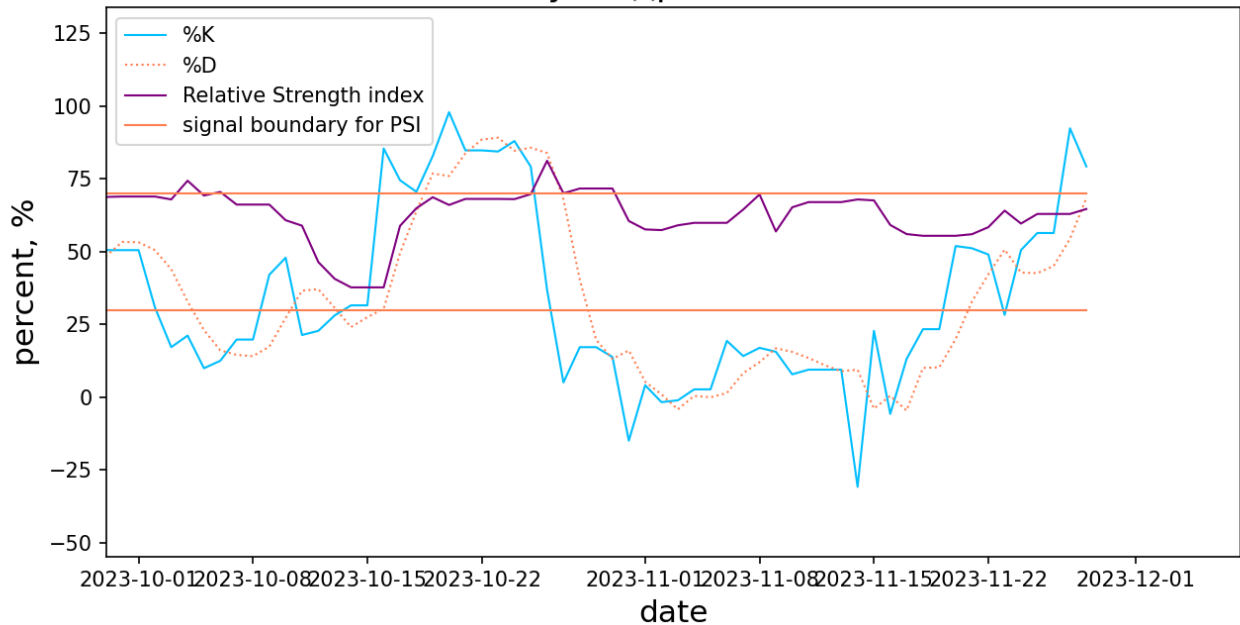
Аллигатор спит, когда три средние переплетены вместе и движутся в узком диапазоне.

Если же сверху расположена зеленая линия (губы аллигатора), а скользящие направлены вверх, то перед вами восходящая тенденция.

Стохастик

Зоны осциллятора выше 80% и ниже 20% называются зонами перекупленности и перепроданности соответственно. Когда бумага входит в зону перекупленности (то есть, торгуется вблизи верхней границы диапазона) если она отскочит вниз, то можно рассчитывать на дальнейшее падение. Сигналом на вход в короткую позицию является пересечение уровня 80% линией %K сверху вниз. Аналогичным образом, сигналом на покупку является пересечение уровня 20% линией K% снизу вверх.

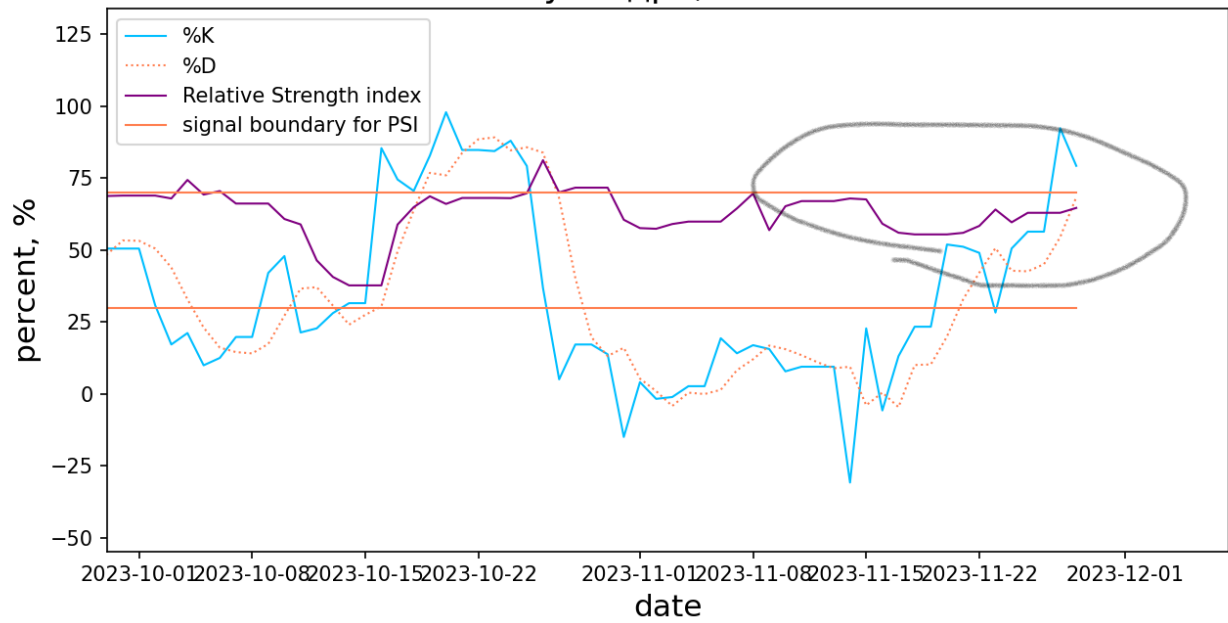
РусГидро, NYDR



В нашем случае пересечения нет, поэтому сигналов не видно

Индекс относительной силы (RSI)

РусГидро, HYDR



В зону перекупленности не входит

Итог: покупать