**Описание алгоритма**

**Цель проекта: создать автоматизированную торговую систему с возможностью добавления торговых инструментов, изменением параметров и управлением через телеграмм бота.**

Основным модулем исполнения торговой системы и анализа исторических данных служит BackTrader (https://www.backtrader.com/).

Загрузка котировок происходит с export.finam.ru для всех основных таймфреймов (см. finam.py) и большого перечня тикеров (см. finam.py) на выбор пользователя. Полученные котировки сохраняются в csv формат и далее подгружаются в BackTrader. Частота запросов каждую минуту. Далее торговая система сама пропускает необходимые данные в нерабочее время и т д.

Торговая система основана на торговли по тренду (при нахождении ценового ряда выше индикатора тренда SMA с периодом 230) и входу в сделку при откате (при нахождении ценового ряда ниже индикатора Lowest с периодом 4), сопровождение - трейлинг стоп с размером 2 ATR за 14 периодов. Совершается не более одной сделки в день. Инструмент - акции Сбербанка. Таймфрейм - часовой. Причины выбора данной торговой системы см. ниже.

Предложенное решение запускает торговую систему через тг-бота. Для проверки инфраструктуры есть стратегия покупки/продажи актива каждую минут. Далее есть возможность остановить торговую систему и закрыть открытые позиции и стоп заявки. Проверить баланс, наличие открытых позиций и получить эти данные в тг-бота. Запустить бэктест стратегии на выбранном таймфрейме и получить данные анализа стратегии, так же в тг бота.

По ходу проекта в коде даны комментарии

**Структура проекта:**

Config.py – основные настройки для тг бота, счета Финам, и пути для сохранения данных

main.py – основной управляющий файл тг бота

Cerebro.py – запуск BackTrader в отдельном потоке

Cerebro\_Analayzer.py - запуск BackTrader в режиме анализа исторических данных в отдельном потоке

finam.py – файл для получения котировок от export.finam.ru

BackTraderFinam – папка с хранилищем (FNStore.py) и брокером (FNBroker.py) для подключение к брокеру Финам

Data – папка с сохраненными файлами данных: скаченные котировки, лог работы проекта (log.log), счетчик сделок (counter.txt)

FinamPy – папка для работа с сервером TRANSAQ через REST/gRPC

Strategy – папка со стратегиями, две стратегии на выбор пользователя Buy\_Sell продажа и покупка актива каждую минуту для проверки инфраструктуры, Trend\_PullBack «боевая» торговая система.

trading\_bot.conf - конфигурационный файл для запуска в supervisor, позволяет работать в фоне и перезапускать проект, необходим для установки на выделенный сервер

Необходимые библиотеки в requirements.txt

**Запуск:**

Изменить настройки в Config.py – получить токен тг бота , ид и название аккаунта для управления, вставить номер счета и торговый токен от Common.

Запуск проекта под windows: python main.py

Запуск проекта под linux (запускал на Ubuntu 22.04): python3 main.py

Запуск проекта под linux через supervisor:

apt-get install supervisor –y

скопировать trading\_bot.conf в /etc/supervisor/conf.d

sudo supervisorctl reread

sudo supervisorctl update

Можно запускать торговые системы и не через тг бота, примеры вызова закомментированы внизу в файлах (Cerebro.py и Cerebro\_Analayzer.py).

Примеры команд (используемые пробелы и обозначения важны):

/start 1 SBER hour 230 4 – запуск основной стратегии на акциях Сбербанка

/start 2 GAZP min – запуск стратегии Покупка/Продажа на акциях Газпрома

/analyze SBER hour 2022-01-01 2023-02-01 230 4 – запуск анализа стратегии на акциях Сбербанка

Остальные команды в /help описаны

**Торговая система (на основе алгоритма 1000 обезьян)**

Выбор торговой системы основан на анализе исторических данных. Ниже на графике можно увидеть интересное свойство акций Cбербанка. При использовании случайного входа и случайного выхода с вероятностью входа 50%, и средней величине вероятности выхода 50% при средней величине длины трейда 5 баров и 1000 прогонов данного алгоритма (аналог торговли 1000 счетов (обезьян) которые совершают случайные сделки) у нас нет убыточных счетов (обезьян).

Результат торговли 1000 счетов (обезьян)

Далее таким же образом задавался не случайный вход (с использованием индикаторов тренда и отката) и выход (трейлинг стоп с определенной величиной), анализировался лучший таймфрейм. Оптимизировались параметры индикаторов, при этом стараясь сделать систему надежной и избежать переоптимизации. Более подробно об этом я могу рассказать в финале.

**Возможные ошибки. О**дин раз возникла ошибка при загрузке данных с export.finam.ru, сервер не отвечал. Возможны задержки со стороны телеграмма, несколько раз были ошибки превышения количества подключений, по всей вероятности это ошибка используемой библиотеки. Один раз была длительная ошибка gRPC соединения.

В целом нужно время для работы, инициализация стратегии идет порядка 5 минут. Остановка потока около 2 минут. Т.е. алгоритм работает не мгновенно.

Проверил все заявки и работу стратегий, по идее все должно работать.