

# Серия 1: инфраструктура для альф.

Дедлайн: 25 февраля.

**Задача 0.** Профильтровать данные. Найти 11 случаев, когда цена уменьшается за 1 день примерно в  $k$  раз, а потом возрастает в  $k$  раз. Будет 9 случаев для  $k = 10$  и 2 случая для  $k = 100$ .

Задачи 1 - 8 нужно сделать для альфа-вектора:

**Задача 1.** Написать операцию нейтрализации.

**Задача 2.** Написать операцию нормализации.

**Задача 3.** Написать операцию truncate.

**Задача 4.** Написать функцию подсчета доходности (return) за день.

**Задача 5\*.** Написать операцию rank (ранжирование).

**Задача 6\*.** Написать операцию CutOutliers (удаление выбросов).

**Задача 7\*.** Написать операцию CutMiddle (удаление средних).

**Задача 8\*.** Написать операцию ApplyFunction (применение передаваемой функции).

Задачи 9 - 14 нужно сделать для матрицы, составленной из альфа-векторов:

**Задача 9.** Написать функцию подсчета turnover с возможностью вывода как вектора значений за все время, так и средних значений по годам (это вывод по умолчанию).

**Задача 10.** Написать функцию подсчета drawdown с выводом значений по годам с возможностью вывода тех дней, когда он достигался.

**Задача 11.** Написать функцию подсчета доходности (return) с выводом за все время (по умолчанию) и совокупные по годам.

**Задача 12.** Написать функцию подсчета коэффициента sharpe с выводом значений по годам.

**Задача 13.** Написать операцию decay (замедление).

**Задача 14.** Написать операцию AlphaStats, которая рисует график доходности за всю историю и выводит за каждый год: sharpe, средний turnover, суммарную доходность за год, drawdown за год. Итого 1 график и 20 чисел.

**Задача 15.** Напишите функцию корреляции между двумя векторами доходности двух альф.

**Задача 16\*.** Напишите проверку того, что у альфы нет forward looking bias — заглядывания в будущие данные.

## Серия 2: Альфы.

Дедлайн: 25 февраля.

Не забудьте проверять все данные на ошибки!

В задачах 1 — 6 нужно сделать альфы, используя указанные факторы в качестве идей, но не ограничиваясь ими. Если фактор из условия сработал плохо, надо его улучшить, переписав в другом виде. Используйте улучшающие множители и операции (задачи 5-8 из серии 1).

**Задача 1.**  $(close(d - n) - close(d - 1)) / close(d - 1), \quad close(d - n)/close(d - 1).$

**Задача 2.**  $high - low, \quad high/low.$

**Задача 3.**  $high + low - 2close, \quad (high + low)/2 < close ? 1 : -1, \quad high * low / close^2$

**Задача 4.**  $open/close, \quad high/close, \quad (high - low)/close.$

**Задача 5.**  $(close - low)/(high - low), \quad (close - low)/(high - low) < 0.5 ? 1 : -1.$

**Задача 6.**  $correlation(volume, close), \quad correlation(high - low, return).$

**Задача 7.** Проверьте, какие из найденных альф (включая слабые) улучшаются с помощью множителя  $ts\_rank(volume, 20)$ ?

**Задача 8.** Является ли  $ts\_rank(returns, 20)$  альфой или улучшающим множителем?

**Задача 9\*.** Сделать альфу с помощью метода KNN (к ближайших соседей).

**Задача 10\*.** Сделать альфу типа price momentum.

## Серия 3: Портфели.

Дедлайн: 2 марта.

Перед построением портфелей посчитайте попарные корреляции между альфами и выведите их в виде матрицы. Выберите топ-3 альфы по доходности и выводите график доходности портфелей вместе с ними. У каждого портфеля нужно посчитать все те же статистики, что и у альф, для этого применяется AlphaStats.

В задачах 1 — 7 нужно построить портфели с указанными весами.

**Задача 1.** Все веса равны.

**Задача 2.** Вес альфы пропорционален ее sharpe.

**Задача 3.** Вес альфы пропорционален ее return.

**Задача 4.** Вес альфы пропорционален ее sharpe или return и обратно пропорционален ее «корреляциям с остальными альфами». Придумайте формулу (их на самом деле несколько), как по набору корреляций с остальными альфами получить один множитель.

**Задача 5.** Веса получите оптимизацией Sharpe по Марковицу.

**Задача 6.** Веса получите по Risk Parity.

**Задача 7.** Веса получите по Risk Budgeting, где назначаемый вклад в общий риск пропорционален return или sharpe альфы.

**Задача 8\*.** Постройте 10000 альф и сделайте для них линейный по альфам аналог оптимизации по Марковицу.

**Задача 9\*.** Решите задачу 8, используя netting добавляемой альфы на каждом шаге.