Московский Государственный Университет имени М. В. Ломоносова

Курсовая Работа

Кафедра: Математических методов анализа экономики

Тема: «Анализ алгоритмов автоматической торговли на рынке ценных бумаг.»

Руководитель ______________________ Е. Н. Лукаш

Выполнил студент группы э – 201 _______________________С. Д. Петраков

Москва

2019

Оглавление

Введение.

Проблема внедрения системы и осуществления автоматической торговли на рынке ценных бумаг.

Автоматическая торговля является системой торговли, созданной на основе алгоритмов, с чем связаны её достоинства и недостатки. Огромное количество достоинств будет освещено ниже в течение всей работы, но самым большим недостатком автоматической торговли является сложность этого процесса, для многих людей сами по себе эти процессы непонятны. Поэтому проблема внедрения системы автоматической торговли стоит особенно остро. Во многом это обуславливается комплексным подходом к этому вопросу. Для реализации данного механизма необходима бесперебойная работа всех его частей, и если что-то не работает, то и вся система рушится, поэтому необходимо отслеживать и проверять все части системы автоматической торговли.

Общая информация и сведения об алгоритмической и автоматической торговле.

«Если хочешь овладеть некоторой областью знаний в совершенстве – обучи этому компьютер». Думаю, что эта фраза очень хорошо раскрывает суть вещей. Несколько слов здесь я отведу автоматической и алгоритмической торговле.

Часто люди путают эти понятия, поэтому для начала стоит их разграничить. Алгоритмическая торговля или алгоритмический трейдинг - это способ исполнения большой заявки (часто имеет место ситуация, когда целиком всю заявку не реализовать за раз, и она делится на более мелкие части) при помощи особых алгоритмов большая заявка (*parent order*) делится на некоторое количество под-заявок (*child orders*), далее каждой из них присваивается собственное значение объёма и цены, а также время исполнения на рынке. Смысл алгоритмической торговли заключён в упрощении торговли и механизации некоторых технических действий, например деление заявки на части. Перед алгоритмической торговлей нет цели получения прибыли. Целью алгоритмической торговли является уменьшение стоимости исполнения крупной заявки (*transaction cost*), минимизация рыночного влияния (*market impact*) и уменьшение риска исполнения заявки.

В отличие от алгоритмической торговли, автоматическая торговля ставит перед собой цель получения прибыли. Второе название автоматической торговли – “торговые роботы” (“black box traiding”). Основа которых заключена в быстрой обработке данных и продвинутой математике. Есть бесчисленное множество разных стратегий воплощения автоматической торговли, о которых я подробно поговорю дальше.

Особенное место в современном торговом мире занимает высокочастотная торговля, которая является ярким предметом для исследования[1] и использования на практике[2], однако в силу своей сложности я лишь затрону теоретическую часть высокочастотной торговли.

Актуальность внедрения автоматической торговли на рынках ценных бумаг.

В современном мире существует чёткая тенденция к автоматизации, роботизации многих процессов, торговля ценными бумагами не исключение. Конечно, всегда существует выбор торговли основанной на фундаментальных воззрениях инвестора, и применение роботизации в этом направлении затруднено, зато огромное поле деятельности можно найти в сфере технического направления, оно как раз сильно совершенствуется. Растущие вычислительные мощности позволяют быстрее анализировать ситуацию, выполнять стратегические команды, все эти новшества дают преимущество тем, кто может ими воспользоваться. Поэтому внедрение систем автоматической торговли становится всё более и более значимым. Согласно исследованиям, проведёнными учёными были достигнуты важные результаты: Полученные данные свидетельствуют о том, что алгоритмическая торговля (AT) увеличивает ликвидность и повышает информативность котировок[3]. Также анализ показал, что AT потребляет ликвидность, когда она дешёвая, то есть, когда разница между котировками спроса и предложения мала (дешёвая), и предоставляет ликвидность, когда она велика (дорогая). При малых спредах AT реже подают новые ордера, реже отменяют свои ордера и чаще инициируют торговлю. AT быстрее реагирует на события.[4] Как можно заметить многие исследователи занимаются

---

[1] Brogaard, J., Hendershott, T., & Riordan, R. (2014). High-frequency trading and price discovery. *The Review of Financial Studies*, *27*(8), 2267-2306. https://doi.org/10.1093/rfs/hhu032

[2] Carrion, A. (2013). Very fast money: High-frequency trading on the NASDAQ. *Journal of Financial Markets*, *16*(4), 680-711. https://doi.org/10.1016/j.finmar.2013.06.005

[3] Hendershott, T., Jones, C. M., & Menkveld, A. J. (2011). Does algorithmic trading improve liquidity?. *The Journal of Finance*, *66*(1), 1-33. https://doi.org/10.1111/j.1540-6261.2010.01624.x

[4] Hendershott, T., & Riordan, R. (2013). Algorithmic trading and the market for liquidity. *Journal of Financial and Quantitative Analysis*, *48*(4), 1001-1024. https://doi.org/10.1017/S0022109013000471

вопросами АТ в данный момент, получают результаты, которые свидетельствуют о структурном изменении торговой среды, которая сказывается здесь и сейчас на всех участниках рынка. Так, например, в статье “Rise of the machines: Algorithmic trading in the foreign exchange market”[5] рассматривается влияние алгоритмической торговли (АТ) на валютный рынок с использованием временных рядов высокочастотных данных, позволяющих идентифицировать активность торговли с помощью компьютера. Авторы приходят к выводу, что АТ приводит к улучшению двух показателей эффективности цен: частоты треугольных арбитражных сделок и автокорреляции высокочастотных доходностей. Показывается, что сокращение арбитражных возможностей связано, прежде всего, с изъятием компьютерами ликвидности. Этот результат согласуется с мнением, что АТ повышает информационную эффективность, ускоряя обнаружение цен, но может также приводить к более высоким затратам для трейдеров. Снижение автокорреляции доходности, напротив, в большей степени связано с алгоритмическим обеспечением ликвидности.

Статьи, посвящённые тематике влияния алгоритмической и автоматической торговли весьма молоды, что говорит о том, что они не только представляют практический интерес для торговли на рынке ценных бумаг, но и привлекают исследовательский и академический интерес для изучения последствий тех структурных изменений, которые имеют место.

Описание подходов и стратегий, рассмотренных в работе.

Я отведу стратегиям центральное место в работе, рассмотрю их широкий спектр, вообще говоря разделенный на два крупных лагеря: возврата к среднему значению (mean-reverting strategies) и импульсные стратегии (momentum strategies), рассмотрю стандартные методы торговли каждой стратегии, что не менее важно, фундаментальные причины, по которым стратегия должна работать. Акцент будет поставлен на линейных стратегиях. Рассмотрим ключевые моменты трейдинга: коэффициент Шарпа, величину оптимального кредитного плеча (Леверидж) на базе формулы Келли.

5. Chaboud, A. P., Chiquoine, B., Hjalmarsson, E., & Vega, C. (2014). Rise of the machines: Algorithmic trading in the foreign exchange market. *The Journal of Finance*, *69*(5), 2045-2084. https://doi.org/10.1111/jofi.12186

Обсудим несколько статистических методов (расширенный тест Дики - Фуллера (ADF test), расширенный коинтеграционный тест Дики – Фуллера (CADF test) для выявления временных рядов, их преимущества в использовании, имеющих свойство возвращения к среднему значению, обоснование стационарности временного ряда, показатель Хёрста (Herst exponent), тест дисперсионного отношения (Variance Ratio Test), период возврата средней реверсии (возвращение), тест Йохансена (Johansen test)[6]. Помимо механического применения этих статистических тестов к временным рядам, передадим интуитивное понимание того, что они на самом деле тестируют, и простые математические уравнения, стоящие за ними. Объясним методы и стратегии торговли портфелями со средней реверсией (возвращением) (линейную). Отведём немало места вопросом оптимального хеджирования. Также рассмотрим вопрос того, имеют ли цены на сырье, логарифмические цены или коэффициенты отношения цен смысл в качестве входных данных для этих тестов и стратегий. Рассмотрим различные вариации и комбинации перечисленных методов и стратегий, опасности ошибок в данных, важность спредов, и особенную опасность в данных для этого случая. Рассмотрим примеры с различными ценными бумагами, индексами, биржевыми инвестиционными фондами (ETF), а также их комбинациями, рассмотрим парную торговлю, торговлю трио и более компонентов. Обратим внимание на различие торговых стратегий по временным горизонтам, например внутридневная, меж дневная, долгосрочная. Стоит сразу оговорить тот факт, что в последнее время бывает сложно добиться прибыли в разных ситуациях из-за растущей конкуренции высокочастотной торговли.

При обсуждении импульсных стратегий рассмотрим тесты их выявления, также их объяснения. мы начнем с объяснения нескольких статистических тестов для импульса временного ряда. Рассмотрим плюсы и минусы стратегий импульса в сравнении со стратегиями возвращения к среднему значению. Обратим большое внимание на имплементацию торговых стратегий на примерах, реализованных по большей части на языке Python. Проведём бэктестинг, продемонстрируем анализ полученных результатов.

Технически процесс автоматической торговли таков:

1. Создание стратегии.
2. Сбор рыночных данных.
3. Бэктестирование на собранной информации (при необходимости).

---

[6] Johansen, S. (1988). Statistical analysis of cointegration vectors. *Journal of economic dynamics and control*, *12*(2-3), 231-254. https://doi.org/10.1016/0165-1889(88)90041-3

4. Отправка заявки брокеру через сервис API и получение уведомления о заявки от брокера.

Содержание моей работы будет освещать пункты 1-3.

Приведение примеров и описание различных типов стратегий.

Подробно остановлюсь на стратегии возврата к среднему значению, а также на типе моментной стратегии[7], в статье приводится пример не только отдельного использования этих глобальных подходах, но и на их комбинации в рамках одной стратегии. Начнём со стратегии средней реверсии:

В нашей жизни многие вещи имеют тенденцию к повторению, возврат к чему-то можно замечать повсеместно, примерами служат сезонные природные явления, вроде изменения средне - суточной температуры, разливов рек и т.д. При анализе таких явлений принято говорить о временных рядах. Если бы средняя реверсия (mean reversion) или, что то же самое, что и стратегии возвращения к среднему значению, встречалась сплошь и рядом в финансовых временных рядах, то жизнь была бы очень проста, потому что достаточно было бы дождаться пока цена опустится ниже (или поднимется выше) своего среднего значения, а затем купить (продать) актив. Однако в действительности большинство временных рядов имеют свойство невозвращения к среднему, а описываются случайными блужданиями. Те немногие временные ряды, которые имеют свойство возвращения к среднему называются стационарными. Теперь перед нами стоит вопрос, как определить, стационарен временной ряд или нет. Для ответа на этот вопрос применяются различные статистические тесты, которые я освещу ниже. Среди них:

- Расширенный тест Дики – Фуллера (ADFtest)
- Показатель Хёрста (Herst Exponent)
- Тест дисперсионного отношения (VarianceRatioTest)
- Период полураспада (Half-life)

Более того, можно обнаружить ещё больше временных рядов, имеющих свойство возврата к среднему значению, если объединить два или три отдельных временных ряда, которые в свою очередь вовсе не обязаны обладать свойством возврата к среднему значению. Временные ряды, которые обладают вышеописанным свойством будем называть

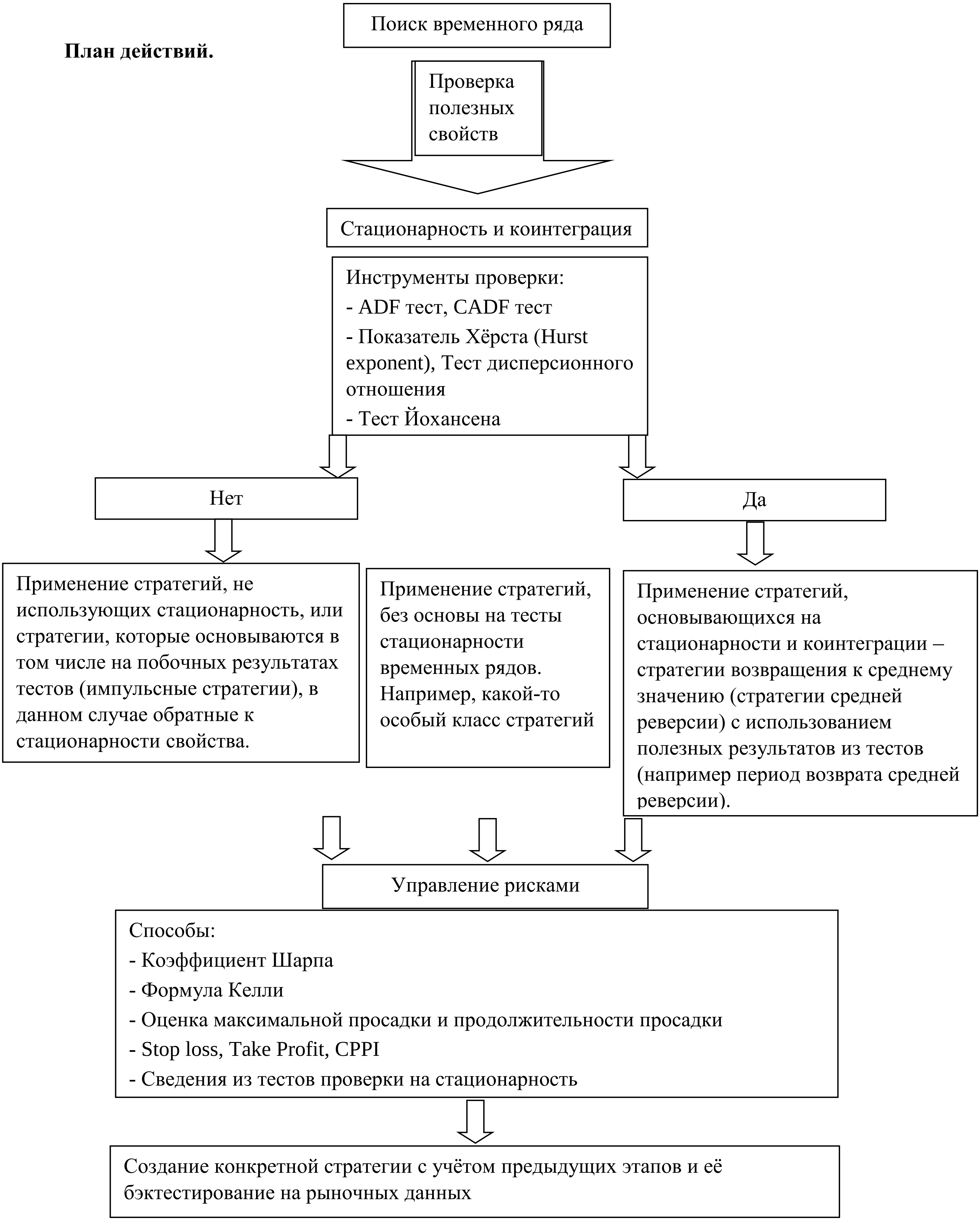
[7] Balvers, R. J., & Wu, Y. (2006). Momentum and mean reversion across national equity markets. *Journal of Empirical Finance*, *13*(1), 24-48. https://doi.org/10.1016/j.jempfin.2005.05.001

коинтеграционными. Для коинтегрирования временных рядов применяются такие тесты как:

- Коинтеграционный расширенный тест Дики – Фуллера (CADFtest)
- Тест Йохансена (Johansen test)

А также в качестве дополнения к тесту Йохансена можно определить точные веса каждого актива для создания портфеля со свойством возвращения к среднему.

План действий.



Глава 1. Теоретические основы алгоритмов автоматической торговли.

1.1. Концепция.

Введём ряд понятий:

1. **Стационарность.**

   Ряд y(t) называется **стационарным** в узком смысле или строго стационарным (strictly stationary), если совместное распределение m наблюдений $y_{t1,}y_{t2}, \ldots, y_{tm}$ не зависит от сдвига по времени - совпадает с распределением $y_{t1+t,}y_{t2+t}, \ldots, y_{tm+t}$ для любых $m, t, t_1, t_2, \ldots, t_m$.

2. **Коинтеграция.**

   Положим, есть нестационарный ряд x_t. Возьмём его первые разности $\Delta x_t = x_t - x_{t-1}$. Если ряд Δx_t является стационарным, то x_t называется интегрируемым порядка 1 (integrated order 1), I(1). Соответственно, стационарны ряд Δx_t называется I(0). Вообще, ряд называется интегрируемым порядка k, I(k), если он и его разности до порядка k-1 включительно нестационарны, а k-я разность стационарна. Пуст теперь есть два I(1) ряда, x_t и y_t. Пусть, кроме того, их линейная комбинация и $y_t - \beta x_t$ является стационарной, I(0). В этом случае ряд x_t и y_t называются **коинтегрированными** (cointegrated), а вектор $(1, -\beta)^I$ называется коинтегрирующим вектором.

3. **Средняя реверсия** = стратегия возвращения к среднему значению. (Для сокращения иногда писать коротко)

Основной концепцией является понятие стационарности и коинтеграции, если речь идёт о средней реверсии. Нашей главной задачей является создание такого объекта (временного ряда) из имеющихся рыночных данных, и которым в итоге можно будет торговать, поведение которого поддаётся анализу и прогнозу. Создать такой предсказуемый временной ряд помогают базовые распространённые стратегические подходы, а проверка на нужные нам свойства обычно происходит с помощью тестов. Я говорю именно временной ряд, а не цена, потому что результат нашей работы – это как правило набор рыночных инструментов, взятых в определённой пропорции и на определённое время. Все пункты создания такого временного ряда будут не раз продемонстрированы и не одним способом.

1.2. Коэффициент Шарпа.

Колоссальное значение имеет данный показатель, поэтому я отведу ему отдельное место.

Определение. Коэффициент Шарпа — это коэффициент, который оценивает эффективность инвестиционного портфеля.

Коэффициент вычисляется путём деления средней премии за риск на стандартное отклонение портфеля.

$$S = \frac{M[R - R_f]}{\sigma} = \frac{M[R - R_f]}{\sqrt{D[R - R_f]}}$$

Где R – доходность портфеля, R_f –доходность от альтернативного вложения (обычно полагают безрисковой процентной ставкой), $M[R - R_f]$ – математическое ожидание, σ – стандартное отклонение доходности портфеля.

Коэффициент Шарпа применяется для анализа того, насколько доходность портфеля компенсирует риск. Практически это означает, что при рассмотрении двух портфелей с одинаковой доходностью и разными коэффициентами Шарпа, предпочтение будет отдано портфелю с большим коэффициентом Шарпа, поскольку этот портфель менее рискован.

Определение. Просадка (англ. drawdown) – это плавающий или реальный убыток на торговом счете трейдера, оцениваемый в процентах или цифрах.

В этой части следует упомянуть о просадках, так как при больших и продолжительных просадках шансы того, что коэффициент Шарпа высок малы. Вообще говоря, любая стратегия, имеющая коэффициент Шарпа меньше единицы обычно неприемлема в качестве самодостаточной стратегии. Для стратегии, которая приносит прибыль почти каждый месяц, годовой коэффициент Шарпа больше двух, для стратегии, которая является прибыльной почти ежедневно, коэффициент Шарпа обычно больше трёх. Для каждого примера я буду приводить коэффициент Шарпа (где это актуально), поскольку этот показатель важен, некоторые трейдеры ставят для себя в некоторых ситуациях ограничение снизу на данный коэффициент. В некоторых случаях по этому коэффициенту можно репрезентативно сравнить разные стратегии: существуют отдельные тесты, основу которым полагает коэффициент Шарпа, эту ситуацию рассматривает научная статья[8].

[8] Ledoit, O., & Wolf, M. (2008). Robust performance hypothesis testing with the Sharpe ratio. *Journal of Empirical Finance*, *15*(5), 850-859. https://doi.org/10.1016/j.jempfin.2008.03.002

1.3. Управление рисками.

На самом деле для разных людей управление рисками значит разные вещи. Обычный человек испытывает “отвращение” к потерям, в англоязычной литературе принято говорить “risk averse”. Иными словами, для компенсации одного доллара потери нужно, например, 2 доллара (здесь заключается связь с коэффициентом Шарпа: его значение на уровне 2 кажется привлекательным).

Вообще говоря, такой подход нерационален, так как мешает максимизировать рост долгосрочного акционерного капитала, что является основной задачей.

Ключевой концепцией управления рисками является разумное использование заёмных средств, количество которых мы можем оптимизировать с помощью формулы Келли или некоторых численных методов, которые максимизируют совокупные темпы роста.

Также можно использовать stop loss. Другой путь - страхование портфеля с постоянной пропорцией.

Наконец, было бы разумно вообще избегать торговли в то время, когда риск потери велик. Рассмотрим некоторые опережающие индикаторы.

Оптимальное кредитное плечо.

Определение. Кредитное плечо (Леверидж) – это заимствование средств для увеличения доходов от инвестиций (соотношение собственных и заёмных средств).

Легко сказать, что мы должны быть осторожными. Труднее решить, что является разумным или оптимальным рычагом воздействия для конкретной стратегии или портфеля, поскольку, очевидно, если мы установим леверидж на нуле, мы не пострадаем от рисков, но и не получим дополнительной прибыли.

Нас интересует такой кредитный рычаг, который максимизирует совокупный темп роста. Каждый метод имеет свои собственные особенности и недостатки.

Важным сходством является то, что во всех случаях мы исходим из того, что распределение вероятности получения прибыли рынка в будущем аналогично прошлому. Обычно это неверное предположение, но это лучшее, из того, что может показаться неправильным. Мы предполагаем, что распределение вероятностей доходности стратегии является гауссовым.

Как это часто бывает в математическом моделировании, наиболее ограничительные допущения приводят к наиболее элегантному и простому решению.

Начнём с формулы Келли с предположением о гауссовском распределении доходности. Важным моментом в анализе рисков является максимальный размер просадки (MaxDrowdown).

Начнём с того, что независимо от того, как будет определено оптимальное кредитное плечо, ключевым фактором является то, что рычаги воздействия должны быть постоянными. Это необходимо для оптимизации темпов роста независимо от того, есть ли у нас максимальное ограничение просадки или нет. Поддержание постоянного рычага воздействия может звучать довольно обыденным, но может противоречить интуиции, при реализации: простыми словами это можно охарактеризовать, как настойчивость, постоянство инъекций, даже в случае убыточности (если не предусмотрено иного: stop loss, например)

Формула Келли.

Положим, что доходность ценной бумаги – это случайная величина, имеющая гауссовское распределение, тогда согласно формуле Келли оптимальный леверидж (кредитное плечо) f можно посчитать по формуле:

$$\mathrm{f} = \frac{\mathrm{m}}{s^2}$$

где m – средняя избыточная доходность (разность доходности актива и доходности безрискового актива), s^2–дисперсия избытка доходности (дисперсия разности доходности актива и доходности безрискового актива).

Вывод формулы Келли в предположении, что происходит гауссово распределение доходности.

Мы начинаем с формулы темпа роста g(f), применительно к Гауссовому процессу по Thorp, E. O. (2011):

$$\mathrm{g(f)} = \mathrm{r} + \mathrm{f\,m} - \frac{s^2\mathrm{f}^2}{2}$$

где f - кредитное плечо, r - безрисковая ставка; m - средняя избыточная доходность, s - стандартный показатель отклонения от избыточной доходности. Чтобы найти оптимальный f, который максимизирует g, просто продифференцируем gпо fи приравняем к нулю (необходимое условие экстремума):

$$\frac{\mathrm{dg}}{\mathrm{df}} = \mathrm{m} - s^2\mathrm{f} = 0$$

Решение этого уравнения для f дает нам $f = \frac{m}{s^2}$.

Предположим, мы планируем торговать несколькими стратегиями, каждая из которых имеет свою собственную среднюю ожидаемая доходность и среднеквадратические отклонения. Как тогда мы должны распределить капитал среди них оптимальным образом? Кроме того, каким должен быть леверидж?

Нашей целью здесь является максимизация нашего долгосрочного богатства. Максимизация долгосрочного благосостояния эквивалентна максимизации долгосрочного совокупного темпа роста g нашего портфеля.

Одно из предположений, которое я сделаю, это то, что распределение доходности каждой из торговых стратегий i - гауссово, с фиксированным средним значением m_i и стандартным отклонением s_i. (предполагаем чистую (избыточную) доходность).

Обычно это неточное приближение, но это лучший вариант, который может быть предложен.

Давайте обозначим оптимальные доли нашего капитала, которые мы должны распределить по каждой из n стратегий, за вектор столбец $F^* = (f_1^*, f_2^*, \dots, f_n^*)^T$.Здесь, T означает транспонирование.

Учитывая нашу цель оптимизации и гауссовское предположение, Доктор Торп (Dr. Thorp)[9] показал, что оптимальным распределением является:

$$F^* = C^{-1}\text{M}$$

Здесь, C - ковариационная матрица, поэтому элемент Cij -ковариация доходностей i - ой и j-ой стратегий, -1 указывает на обратную матрицу, а $M = (m_{1,}m_2, \dots, m_n)^T$–вектор столбецсредних доходностей стратегий.

Если предположить, что все стратегии статистически независимы, то ковариационная матрица становится диагональной, с элементами по диагонали, которые равны дисперсиям отдельных стратегий. Из этого следует, что можно использовать формулу Келли:

$$f_i = \frac{m_i}{s_i^2}$$

Часто, из-за неопределенности в оценках параметров, а также и потому, что распределение прибыли не совсем гауссово, трейдеры предпочитают сократить это

[9] Thorp, E. O. (2011). The Kelly criterion in blackjack sports betting, and the stock market. In *The Kelly Capital Growth Investment Criterion: Theory and Practice* (pp. 789-832). https://doi.org/10.1016/B978-044453248-0.50015-0

рекомендованное плечо наполовину для безопасности. Это называется “половиной ставки на Келли".

Я заявил, что принятие такого распределения капитала и левериджа позволит нам максимизировать долгосрочные совокупные темпы роста благосостояния. Так что же это за максимальные темпы роста? Оказывается, это значение равно

$$g = r + \frac{S^2}{2}$$

где S – это коэффициент Шарпа в нашем портфеле.

Заметим, что чем выше коэффициент Шарпа в нашем портфеле, тем выше максимальные темпы роста нашего капитала, при условии, что мы используем рекомендованные оптимальные рычаги воздействия по формуле Келли. Это простое математическое воплощение этого факта.

Максимальная просадка (Maximum Drawdown).

На практике трейдеру важно, насколько велика наибольшая просадка, а также наибольший период спада, это нужно для планирования своей стратегии. То есть обычно ставится некоторый приемлемый максимум для этих значений. Пример вычисления продолжительности и величины просадки продемонстрирую в части Практика.

Страхование портфеля с постоянной долей (Constant Proportion Portfolio Insurance).

Страхование портфеля с постоянной долей (CPPI) – это вид портфельного страхования, при котором инвестор устанавливает минимальную долларовую стоимость своего портфеля, а затем структурирует распределение активов в соответствии с этим решением. В CPPI используются два класса активов: рискованный актив (обычно акции или паевые инвестиционные фонды) и консервативный актив в виде денежных средств, их эквивалентов или казначейских облигаций. Процентное соотношение зависит от величины "подушки", определяемой как (текущее значение портфеля минус минимальное значение), и коэффициента мультипликатора, где более высокое значение означает более агрессивную стратегию.

Работает это следующим образом:

Инвестор осуществит начальную инвестицию в рискованный актив на сумму, равную стоимости: (Мультипликатор) * (стоимость «подушки») и будет инвестировать остаток в консервативный актив. По мере изменения стоимости портфеля с течением времени, инвестор будет менять баланс в соответствии с той же стратегией.

Рассмотрим гипотетический портфель в 100 000 долларов, из которых инвестор решает, что 90 000 долларов является абсолютным минимумом. Если стоимость портфеля упадет до $90 000, инвестор переместит все активы в наличные для сохранения капитала.

Величина мультипликатора основана на профиле риска инвестора и определяется по первому вопросу: какой максимальный однодневный убыток может быть в случае рискованной инвестиции. Коэффициент будет обратным этому проценту. Таким образом, если принять решение о том, что 20 процентов — это максимальная вероятность просадки, то значение множителя будет равно (1/0.20), то есть 5. Значения мультипликаторов между 3 и 6 очень распространены.

Основываясь на предоставленной информации, инвестор выделяет 5 x (100 - 90 000 долл. США) или 50 000 долл. США на рискованный актив, а остаток направляется в денежные средства или консервативный актив.

График ребалансировки зависит от инвестора, часто рассматриваются ежемесячные или квартальные примеры. В идеале стоимость подушки со временем будет расти, позволяя больше денег вкладывать в рискованный актив.

Стоп лосс и тейк профит (Stop Loss и Take Profit).

Стоп лосс – это заявка на бирже, которая выставляется в торговом терминале инвестором или трейдером, и целью которой является ограничение убытков, если цена достигает заранее определённого уровня. Тейк профит – это заявка, зеркальная к заявке стоп лосс с целью ограничить выигрыш. Идея заключается в ограничении жадности трейдера им самим.

1.4. Основы стратегий автоматической торговли. Базовые математические модели и тесты временных рядов.

Коинтеграция.

Большинство финансовых временных рядов нестабильны, то есть не обладают свойством возврата к среднему значению. Но, к счастью, мы можем так создать портфель индивидуальных временных рядов, чтобы рыночная стоимость (или цена) этого портфеля обладала свойством возврата к среднему значению. Это и есть понятие коинтеграции: если мы можем найти стационарную линейную комбинацию нескольких нестационарных временных рядов, тогда эти временные ряды называются коинтегрированными. Наиболее распространенной комбинацией является комбинация двух временных рядов: например, покупка и одновременная продажа активов, с соответствующим распределением капитала для каждого актива. Это известная стратегия “парной торговли”. Однако, концепция коинтеграции легко распространяется на три или более активов. Далее я рассмотрю два распространенных коинтеграционных теста: CADF тест и Johansen (Йохансен) тест. Первое подходит только для пары временных рядов, в то время как второе применимо к любому количеству временных рядов.

Возвращение к среднему и стационарность.

Математическое описание временного ряда с свойством возврата к среднему значению заключается в том, что изменение временного ряда в следующем периоде пропорционально разнице между средней ценой и текущей ценой. Как следствие в ADF тесте происходит проверка возможности отвержения нулевой гипотезы о том, что константа пропорциональности равна нулю.

Математическое описание стационарного временного ряда таково, что дисперсия логарифма цен меняется медленнее, чем дисперсия геометрического случайного блуждания. Таким образом их дисперсия является ***сублинейной*** функцией от времени, вместо линейной функции, как в случае геометрического случайного блуждания. Эта сублинейная функция обычно аппроксимируется τ^{2H}, где τ – время разделения двух измерений цен, H–показатель Хёрста, если он ниже 0,5, то временной ряд действительно стационарен, в случае, если показатель равен 0,5, то временной ряд является геометрически случайно блуждающим.

Тест коэффициента дисперсии (The Variance Ratio test) может быть использован для отвержения нулевой гипотезы о том, что показатель Хёрста равен 0,5.

Теоретической основой, заложенной ещё в 1988 году, понятиям средней реверсии можно найти в статье " Mean reversion in stock prices: Evidence and Implications"[10], в которой поднимаются теоретические основы и рассматриваются формальные конструкции средней реверсии и тестирования.

Другие подтверждения того, что стратегии возвращения имеют место не только в теории, но и на практике: в статье Balvers, R., Wu, Y., & Gilliland, E. (2000). В ней можно также наблюдать доходности стратегии возвращения к среднему значению больше, чем доходности от долгосрочной стратегии «купи и держи»[11]. В статье Chaudhuri, K., & Wu, Y. (2003) авторы утверждают, что рынки, которые они исследовали (рынки 17 развивающихся стран) находятся или в состоянии случайного блуждания или в состоянии возвращения к среднему значению.[12]

Расширенный тест Дики – Фуллера (AugmentedDickey-FullerTest (ADF)).

Основа теста заключена в модели: рассмотрим временной ряд $y(t)$, рассмотрим $\Delta y(t)$, как

$$(0)\ \Delta y(t) = \lambda y(t-1) + \mu + \beta_t + \alpha_1 \Delta y(t-1) + \ldots + \alpha_k \Delta y(t-k) + \varepsilon_t,$$

Где $\Delta y(t) \equiv y(t) - y(t-1), \Delta y(t-1) \equiv y(t-1) - y(t-2),$и так далее. Это временной ряд.

ADFтест отвечает на вопрос, верно ли равенство $\lambda = 0$. Если нулевая гипотеза $\lambda = 0$ может быть отвергнута, то получаем $\Delta y(t)$ зависит от текущего уровня $y(t-1)$, следовательно изменение цены не является случайным блужданием.

---

10 Poterba, J. M., & Summers, L. H. (1988). Mean reversion in stock prices: Evidence and implications. *Journal of financial economics*, *22*(1), 27-59. https://doi.org/10.1016/0304-405X(88)90021-9

11 Balvers, R., Wu, Y., & Gilliland, E. (2000). Mean reversion across national stock markets and parametric contrarian investment strategies. The Journal of Finance, 55(2), 745-772.https://doi.org/10.1111/0022-1082.00225

12 Chaudhuri, K., & Wu, Y. (2003). Random walk versus breaking trend in stock prices: Evidence from emerging markets. *Journal of Banking & Finance*, *27*(4), 575-592. https://doi.org/10.1016/S0378-4266(01)00252-7

Тестовой статистикой является коэффициент $\frac{\lambda}{SE(\lambda)}$, где SE(λ)–стандартная ошибка, значения коэффициента затабулированы.

Следует обратить внимание на то, что поскольку ожидается возвращение к среднему значению (mean regression) $\frac{\lambda}{SE(\lambda)}$ должно быть отрицательным, и для отвержения гипотезы отношение должно быть больше по модулю, чем критическое значение. Сами критические значения зависят от размера выборки и того, будем ли мы считать, что временной ряд имеет нулевое среднее значение $-\frac{\mu}{\boldsymbol{\lambda}}$ или постоянное движение $-\frac{\beta t}{\boldsymbol{\lambda}}$. В практической торговле постоянный дрейф цены, если таковой имеется, как правило, имеет гораздо меньшую величину, чем ежедневные колебания цен. Поэтому для простоты будем считать, что этот дрейфовый член равен нулю (β = 0).

Расширенный коинтеграционный тест Дики – Фуллера. (Cointegrated Augmented Dickey-Fuller Test (CADF)).

Как уже упоминалось ранее далеко не все временные ряды стационарны. Чтобы добиться стационарности временного ряда и применяется CADFтест. Основой формирования коинтегрированного временного ряда служит коэффициент хеджирования, который можно получить с помощью CADFтеста. Коэффициент хеджирования конкретного актива — это количество единиц этого актива, которые мы должны иметь в портфеле. Если речь идёт об акциях, то количество единиц соответствует количеству акций. Отрицательный коэффициент хеджирования означает в таком случае короткую продажу акций. То, что набор временных рядов является коинтегрированным не означает, что любая случайная линейная комбинация из них сформирует стационарный портфель. Но, следуя мысли, что, если мы сначала определим оптимальный коэффициент хеджирования путем построения линейной регрессии между двумя временными рядами, а затем используем этот коэффициент хеджирования для формирования портфеля, после чего, наконец, выполним тест стационарности на рядах цен из портфеля. Воплотить вышесказанное в жизнь помогает Python с библиотеками и функцией, позволяющей выполнить CADF тест. В части Практика я рассмотрю пару биржевых фондов (ETF) EWA и EWC и применю к ним CADF тест в качестве примера.

Показатель Хёрста (Hurst exponent) и Тест дисперсионного отношения (ADF test).

На интуитивном уровне “стационарный” временной ряд означает, что ряд изменяется от его начального значения медленнее, чем геометрическое случайное блуждание. Математически мы можем определить природу временного ряда, измерив скорость изменения. Скорость изменения может характеризоваться дисперсией.

(1) $\mathrm{Var}(\tau) = \langle|z(t+\tau) - z(t)|^2\rangle$,

где z– это логарифм цен ($z = \ln(y)$), τ–произвольный временной лаг, $\langle\ldots\rangle$ - среднее значение по всем t.

Известно, что для геометрическое случайное блуждание характеризуется следующей эквивалентностью:

(2) $\langle|z(t+\tau) - z(t)|^2\rangle \sim \tau$,

$\sim$ означает, что это отношение превращается в равенство с некоторой константой пропорциональности для больших τ, но оно может отклоняться от прямой для малых τ. Но если (log) временной ряд означает возврат или тренд (т. е. имеет положительные корреляции между последовательными движениями цены), уравнение (2) не будет выполняться. Вместо этого мы можем написать:

(3) $\langle|z(t+\tau) - z(t)|^2\rangle \sim \tau^{2H}$,

где H - показатель Хёрста для временного ряда, демонстрирующего геометрическое случайное блуждание, в случае H = 0,5. Но для ряда со средним возвратом (mean-reverting) H < 0,5,и для тренда, H > 0,5.По мере приближения показателя Hк нулю, можно с большей степенью утверждать свойство возврата к среднему значению для временного ряда, и, соответственно, при увеличении H к 1, временной ряд все больше и больше усиливается «трендовость»; Таким образом, H служит также как индикатор для степени среднего возвращения или силы тренда.

При практическом вычислении показателя Хёрста для временного ряда курсов валют USD.CAD, получаем H = 0,49, что говорит о том, что цена имеет слабое свойство возврата к среднему.

Подсчёт показателя Хёрста (Hurst exponent).

В общем виде в уравнении для расчёта показателя Хёрста фигурирует дополнительный параметр q, означающий произвольное число:

(4) $\langle |z(t+\tau) - z(t)|^{2q} \rangle \sim \tau^{2H(q)}$

Можно реализовать с помощью функций: H=genhurst(log(y), 2);

Тест дисперсионного отношения.

В силу конечной выборки можно узнать статистическое значение оценочного значения H, чтобы быть уверенным, можем ли мы отклонить нулевую гипотеза, что H действительно 0.5. Этот тест гипотезы обеспечивается Тестом дисперсионного отношения (Variance Ratio test) (Lo, 2001).

Сущность Теста дисперсионного отношения – это проверка на верность равенства

$$\frac{\mathrm{Var}\big(z(t) - z(t-\tau)\big)}{\tau Var\big(z(t) - z(t-1)\big)} = 1$$

Простым языком, мы проверяем, что дисперсия показателя в момент времени t и момент t-1 в τ раз меньше, чем дисперсия показателя в момент времени t и момент t – τ. Если равенство достигается, то говорят, что Тест дисперсионного отношения выполнен.

Период возврата средней реверсии.

Статистические тесты, которые я описал для средней реверсии или стационарности очень требовательны, с их требованиями не менее 90 процентов определенности.

Но в практической торговле часто можно работать с гораздо меньшей уверенностью. В этом разделе мы найдем другой способ интерпретировать коэффициент λ в уравнении (0) так, чтобы можно было ответить на вопрос, достаточно ли он отрицателен, чтобы считать торговую стратегию практичной, даже если мы не можем отклонить нулевую гипотезу о том, что его фактическое значение равно нулю с 90-процентной вероятностью по Тесту ADF.

Итог, λ — это мера того, сколько времени требуется для цены, чтобы вернуться.

Разбор: для того, чтобы раскрыть это новое толкование параметра λ, необходимо преобразовать дискретное уравнение временного ряда (0) к дифференциальной форме так, что изменения в ценах становятся бесконечно малыми. Кроме того, если мы проигнорируем дрейф($ß_t$) и разницу в лагах ($\Delta y(t-1)$, ..., $\Delta y(t-k)$) в уравнение (0). В таком случае уравнение становится формулой Орнштейна-Уленбека в стохастическом исчислении для процесса возврата к среднему значению:

(5) $dy(t) = (\lambda y(t-1) + \mu)dt + d\varepsilon$,

где $d\varepsilon$ - гауссовский шум.

Дополнительным преимуществом записи уравнения в дифференциалах является возможность аналитического решения для ожидаемого значения y(t):

(6) $E\big(y(t)\big) = y_0 \exp(\lambda t) - \frac{\mu}{\lambda}(1 - \exp(\lambda t))$

Вспоминая, что λ отрицательна для процесса возврата к среднему значению, следовательно ожидаемое значение цены убывает экспоненциально до значения $-\mu/\lambda$, при этом период возврата равен $-\ln(2)/\lambda$. Эта взаимосвязь коэффициента регрессии λ и периода возврата средней реверсии очень полезны.

Во-первых, если положительна, это означает, что временной ряд не имеет свойство возвращения к среднему значению, и поэтому стратегия возвращения к среднему не применима для такого временного ряда.

Во-вторых, если λ очень близко к нулю, то это означает, что период возврата будет очень долгим, а стратегия торговли со средним возвратом не будет очень прибыльна, так как в заданный период времени мало перемещений “туда, обратно” будет совершено.

Тест Йохансена (JohansenTest).

Для того, чтобы проверить коинтеграцию более двух переменных, необходимо использовать тест Йохансена. Чтобы понять этот тест, следует обобщить уравнение (0) к случаю, когда ценовая переменные y(t) фактически является векторами, представляющими множественные временные ряды и коэффициенты λ и α на самом деле являются матрицами. (В силу того, что практично допускать постоянный дрейф в цене стационарного портфеля будем считать $\beta t = 0$ для простоты.) С помощью английских и греческих букв для представления векторов и матриц соответственно, мы можем переписать уравнение (0)в уравнение

(7) $\Delta Y(t) = \Lambda Y(t-1) + M + A_1 \Delta Y(t-1) + \ldots + A_k \Delta Y(t-k) + \varepsilon_t$

Как и в одномерном случае, если $\Lambda = 0$, мы не имеем коинтеграции. (Напомним, что если следующее значение Y не зависит от текущего уровня цены, не может быть никакой средней реверсии.) Обозначим ранг матрицыΛ как r, и число временных рядов n.

Количество независимых портфелей, которые могут быть сформированы различными линейными комбинации коинтегрированного временного ряда равны r. Тест Йохансена вычислит r для нас двумя различными способами, основанными на разложении собственных векторов Λ. Один тест производит так называемую статистику трассировки, а другой производит собственные статистики.В сущность этих процессов я углубляться не стану, поскольку функционал Pythonобеспечивает нас критическими значениями для каждой статистики, чтобы мы могли проверить отклонение нулевой гипотезы, что $r = 0$ (то есть не существует допустимой коинтеграции) $r \le 1$, ..., до $r \le n - 1$. Если все эти гипотезы отвергнуты, то получаем$r = n$. В качестве полезного побочного продукта можно использовать найденные собственные векторы, как коэффициенты хеджирования для отдельных временных рядов, формирующих стационарный портфель. Продемонстрирую процесс на примере пары EWA-EWC в практической части. Я покажу, как выполнить этот тест на паре EWA-EWC в части Практика.

1.5. Модели торговли.

Линейная модель торговли по стратегии возврата к среднему значению.

После того, как мы определили, что временной ряд обладает свойством средней реверсии и период возврата к среднему значению входит в наши инвестиционные горизонты можно применить линейную модель торговли по стратегии возврата к среднему значению.

Продемонстрирую модель на примере для лучшего понимания:

В этой простой стратегии мы стремимся владеть рядом единиц USD.CAD, равным отрицательному нормализованному отклонению от его скользящего среднего значения. Рыночная стоимость (в долларах США) одной единицы валютной пары USD.X равняется котировке USD.X, так в этом случае линейная средняя реверсия эквивалентна установке рыночной стоимости в портфеле на уровне отрицательной Z-оценки USD.CAD. (Стандартизованная оценка (z-score) - это мера, которая характеризует относительный разброс измеренного или наблюдаемого значения, показывающая, сколько стандартных отклонений составляет его разброс относительного среднего значения.)

Линейная торговля портфелем по стратегии возврата к среднему значению.

В Практической части мы определили, что портфель EWA-EWC-IGE сформирован с “лучшим” собственным вектором из теста Йохансена имеет самый короткий период возврата к среднему значению.

Теперь можно уверенно приступить к тестированию простую линейную стратегию с обратной реверсией по этому портфелю. Идея такая же, как раньше, когда мы владеем количество единиц в USD.CAD пропорциональна их отрицательному нормализованному отклонению от скользящей средней (т. е. Z-Score). Здесь, мы также аккумулируем единицы портфеля пропорциональны отрицательному Z-Score цены “единичного” портфеля. Блок портфель с акциями определяется тестом Йохансена с помощью собственного вектора.

Заметим, что под “линейной” стратегией мы подразумеваем только то, что количество единиц пропорциональна Z-оценке, а не в том, что рыночная стоимость инвестиций пропорциональны.

Такая линейная стратегия не имеет параметров оптимизации, поэтому её разве что можно тестировать на исторических данных.

Плюсы и минусы стратегий возврата к среднему значению.

Часто довольно легко построить стратегии средней реверсии, потому что мы не ограничены торговыми инструментами, которые по своей сути являются стационарными. Мы можем выбирать из большого разнообразия коинтеграций акций и ETF для создания собственного стационарного, то есть со свойством возврата к среднему значению портфеля. Дело в том, что каждый год создаются новые ETF, которые могут незначительно отличаться от существующих, но, безусловно, помогают и нашему делу.

Помимо множества вариантов, часто есть хорошая фундаментальная история пары со средним возвратом. Почему EWA коинтегрирована с EWC? Потому что и в канадской, и в австралийской экономике доминируют товары широкого потребления. Почему GDX коинтегрирована с GLD? Потому что стоимость золотодобывающих компаний очень сильно зависит от стоимости золота.

Даже когда пара коинтеграции распадается (прекращает сближения), мы можем понять причину. Например, причинами, по которым GDX и GLD развалились в начале 2008 года, были высокие цены на энергоносители, из-за которых добыча золота была аномально дорогой. Остаётся надеяться, что с пониманием ситуации придёт исправление стратегии.

Минусом одного – является достоинство другого, так фундаментальные рассуждения справедливые для стратегии средней реверсии контрастирует со многими стратегиями импульса, единственным оправданием которых является скорость реакции на новость

Ещё одно преимущество стратегий возврата к среднему значению заключается в том, что они выполнимы в разных временных масштабах. С одной стороны, рыночные стратегии опираются на цены, которые означают-возвращаются в считанные секунды. С другой стороны, фундаментальные инвесторы годами инвестируют в недооцененные акции и терпеливо ждут, чтобы их цены вернулись к "справедливой” стоимости.

Маленькие горизонты торговли – это особенно выгодно для трейдеров, так как краткосрочная перспектива означает большее количество сделок в год, что в свою очередь означает большую статистическую достоверность и более высокий коэффициент Шарпа для нашего тестирования на истории и реальной торговли, и, в конечном счете, более высокая совокупная доходность нашей стратегии.

К сожалению, именно кажущаяся высокой согласованность стратегии средней реверсии может привести к ее возможному падению. Как Майкл Девер отметил, постоянство часто усыпляет трейдеров. Проблемой стратегии возврата к среднему значению может явиться внезапный сбой, например, по фундаментальным причинам, и разобраться в котором можно только постфактум, но при этом эти редкие потери, которые могут быть хоть и редкие, но болезненные. Следовательно, управление рисками для стратегии возврата к среднему значению особенно важно.

Краткие итоги на текущий момент:

- Средняя реверсия означает, что изменение цены пропорционально разнице между средней ценой и текущей ценой.
- Стационарность означает, что цены изменяются медленнее, чем геометрическое случайное блуждание.
- Тест ADF предназначен для проверки на реверсию среднего значения.
- Тесты показателя Хёрста и коэффициента дисперсии предназначены для проверки на стационарность.
- Период возврата средней реверсии измеряет, как быстро временной ряд возвращается к своему среднему значению, и является хорошим предсказателем прибыльности или коэффициента Шарпа среднеревертирующей торговой стратегии применительно к данному временном ряду.
- Линейная торговая стратегия здесь означает количество единиц или акций единицы портфеля, которое мы имеем пропорционально отрицательный Z-Score временного ряда из этого портфеля.

- Если мы можем объединить два или более нестационарных временных ряда, чтобы сформировать стационарные портфеля, эти цены называются коинтегрированными.
- Коинтеграция может быть измерена или CADF тестомили тестом Йохансена.
- Собственные векторы, полученные из теста Йохансена, могут быть использованы в качестве коэффициентов хеджирования для формирования стационарного портфеля из входных временных рядов, и собственный вектор с наибольшим собственным значением является коинтегрированным рядом с наименьшим периодом возврата к среднему значению.

Реализация стратегий средней реверсии.

Ранее мы описали статистические тесты для определения является ли временной ряд стационарным и, следовательно, пригодным для торговли средней реверсией. Этот временной ряд может быть рыночной стоимостью одного актива (хотя это редкая ситуация).

Важно, что модели, описанные в работе, не учитывают трансакционных издержек, которые необходимо дополнительно учесть при настоящей торговле.

Научная статья на тему торговой стратегии по мотиву средней реверсии – “ Dynamic Trading with Predictable Returns and Transaction Costs”[13]

Торговля парами с использованием ценовых спредов, логарифмических ценовых спредов и коэффициентов.

Ранее при построении портфеля для реверсивной торговли мы просто использовал рыночную стоимость “единичного” портфеля в качестве торгового сигнала. Эта рыночная стоимость или цена является просто взвешенными суммами составляющих временных рядов, где весами являются коэффициенты хеджирования, которые мы нашли из линейной регрессии или из собственных векторов теста Йохансена:

(8) $y = h_1 y_1 + h_2 y_2 + \dots + h_n y_n$

[13] Gârleanu, N., & Pedersen, L. H. (2013). Dynamic trading with predictable returns and transaction costs. *The Journal of Finance*, *68*(6), 2309-2340. https://doi.org/10.1111/jofi.12080

y–это, по конструкции, стационарный временной ряд, а h_i — это число акций каждого рассмотренного в уравнении типа y_i (при условии, что мы торгуем портфелем акций). В случае всего двух акций это сводится к спреду:

(9) $\text{y} = y_1 - \text{h}y_2$.

(Мы вставили знак минус в уравнение (9), чтобы предвидеть тот факт, что обычно одна акция продаётся, а вторая покупается, поэтому знак hявляется положительным.)

Предположим, что смогли достичь такой коинтерграции логарифмов цен:

(10) $\log(\text{q}) = h_1\log(y_1) + h_2\log(y_2) + \dots + h_n\log(y_n)$

Таким образом, что полученный временной ряд стационарен для некоторого множества h, выводящихся или из уравнения регрессии или из теста Йохансена.

Чтобы разобраться в свойстве уравнения (10) рассмотрим конструкцию (разница по времени):

(11) $\Delta\log(\text{q}) = h_1\Delta\log(y_1) + h_2\Delta\log(y_2) + \dots + h_n\Delta\log(y_n)$

Вспоминая, что $\Delta\log(\text{x}) \equiv \log(\text{x(t)}) - \log(\text{x}(\text{t}-1)) = \log(\text{x(t)}/\text{x}(\text{t}-1)) \approx \Delta\,\text{x}/\text{x}$получаем в правой части уравнения (11) следующую конструкцию:

$$h_1\Delta y_1/y_1 + h_2\Delta y_2/y_2 + \dots + h_n\Delta y_n/y_n,$$

Что есть не что иное, как доходность портфеля, состоящего из пценных бумаг, каждая из которых имеется в количестве h_i.

Но в отличие от коэффициентов хеджирования h в уравнении (8), где они являются количествами акций каждого актива, здесь же мы можем установить рыночную стоимость каждого актива как h. Таким образом, мы можем интерпретировать q как рыночную стоимость портфеля активов с ценами y_1, y_2, ..., y_n и с постоянными весами капиталаh_1, h_2, ..., h_n.В случае трейдера основной проблемой является поддержание необходимых весов для реализации стратегии.

Всё это значит, что стратегия средней реверсии через спреды цен проще, чем использование спредов логарифмов цен, но оба они могут быть теоретически оправданы, если и цена, и логарифмы цен коинтегрированны. Продемонстрирую пример стратегии возвращения к среднему значению на основе ценового спреда, спреда логарифмов цен и коэффициентов в главе Практика.

Опасность ошибок в данных.

Ошибки данных оказывают особенно коварное влияние как на тестирование на истории, так и на выполнение стратегий среднего возврата. Если есть ошибки или "выбросы" в исторических данных, используемых для тестирования на истории, то эти ошибки обычно завышают производительность тестирования на истории стратегии среднего возврата. Например, если фактические торговые цены акции в 11:00, 11:01, и 11:02 были $100, $100 и $100, но исторические данные ошибочно записали их как $100, 110$, $100, тогда стратегия вероятно вошла в короткую позицию по акции в 11:01 ($110), а затем покрыла позицию в 11:02 ($100) и получила чистую, но фиктивную прибыль в 10 долларов. Проблема с ошибочными котировками особенно опасна при торговле парами или другими арбитражными стратегиями, поскольку в них мы часто зависим от различий ценовых котировок. В разнице пары котировки обычно заключена гораздо меньшая величина, чем сами котировки, таким образом, любая ошибка в котировках приводит к гораздо большей процентной ошибке в распространении. Например, если мы торгуем парой акций X и Y, а X имеет истинную цена $100 и Y имеет истинную цену $105, поэтому спред Y−X составляет $5, что может быть слишком мало, чтобы вызвать рыночный ордер на покупку Xи продажу Y. Но если ошибка данных заставляет Y отображать цену в $106, то ошибочный спред становится $6, увеличившись на 20 процентов по сравнению с реальным спредомв 5$, и этого может быть достаточно, чтобы спровоцировать ошибочный ордер на покупку X и продажу Y.

Плохие Тики в данных также вызовут у стратегии импульса (моментной стратегии) для отправки неправильный приказ. Таким образом, они в равной степени являются убыточными для исполнения всех стратегий.

Ключевые моменты на текущем этапе:

- Если мы хотим построить портфель со стратегией средней реверсии с фиксированным номером акций во время торговли, то следует использовать временной ряд для определения коэффициента хеджирования.
- Если мы хотим построить портфель со стратегией средней реверсии с фиксированными рыночными стоимостями для каждого компонента во время торговли, то следует использовать логарифмический временной ряд для определения коэффициента хеджирования.
- Коэффициент отношения, вместо спредов, часто является хорошим индикатором для торговли валютой парой.
- Ошибки данных могут улучшать результаты тестирования на истории стратегий со средним возвратом, но не стратегии импульса.
- Стратегии, основанные на спредах, особенно чувствительны к небольшим ошибкам в данных, будь то тестирование на истории или живая торговля.

Средняя реверсия акций и ETF.

Фондовый рынок является, в некотором смысле, наиболее плодородной почвой для поиска инструментов средней реверсии. Сейчас мы рассмотрены вопросы, касающиеся акций и ETF. Продемонстрирую, что простые стратегии возврата к среднему значению работают лучше для пар ETF и трио. Можно не ограничиваться теми стратегиями, которые описаны ранее, когда мы ищем среднюю реверсию в акциях или ETF.

Индексный арбитраж – поиск коинтеграции акций и ETF акций против фьючерсов или акций с ETF.

Статистические тесты для средней реверсии временного ряда в значительной степени не имеют отношения к перекрёстной средней реверсии. Этот дополнительный вид средней реверсии делает создание любого рода стратегии средней реверсии для акций ещё проще.

Из-за этой легкости определения фигур средней реверсии фондовый рынок привлекает большое количество трейдеров, часто называемых статистическими арбитражёрами. В результате отдача в таких стратегиях, как правило, уменьшена. Мы обсудим несколько простых трюков, которые могут повысить их производительность.

Осложнения в торговле парами.

Парная торговля акциями - первый тип алгоритмической стратегии средней реверсии, которую изобрели институциональные трейдеры, как сообщает, Джерри Бамбергер (Gerry Bamberger) в Морган Стэнли (Patterson,2010). В настоящее время может быть удивительно трудно выжать из неё прибыль.

Если тестировать дневные временные ряды отдельных акций, то они почти никогда не соответствуют определению стационарности. Геометрическое случайное блуждание описывает их поведение довольно хорошо. (Их сезонные и суточные свойства, возвращающие среднее значение, являются частными случаями, которые будут обсуждаться позже.)

Даже если сопоставлять акции каким-то разумным образом (например, Exxon против Chevron или Citibank против BankofAmerica), они редко коинтегрируются на протяжении длительного периода времени. Причиной этого различия является то, что состояние одной компании очень быстро меняется в зависимости от управленческих решения и конкуренции. Дело в том, что если обе компании находятся в одном секторе промышленности, то это совсем не гарантирует схожесть их положения (например AAPL против BBRY). Получается, что нужно хорошо разбираться в состоянии компаний для создания работающего примера.

Если мы торгуем большим количеством пар акций, и происходит случайное падение некоторых пар, то Закон больших чисел будет работать только в нашу пользу. Важно, что помимо фундаментальных проблем с торговлей фондовыми парами существуют ещё и две дополнительные технические трудности:

Существует трудность, о которой не стоит забывать - ограничение короткой продажи. Такая ситуация имеет название короткое сжатие (short squeeze).

Внутридневная средняя реверсия: модель Buy – on – Gap.

Цены на акции следуют геометрическим случайным блужданиям. Но это верно только в том случае, если мы проверим их временные ряды для среднего разворота строго через регулярные промежутки времени (например, используя свои ежедневные закрытия). Наша задача - найти особые условия, такие что среднее возвращение происходит с регулярностью. Следующая стратегия покажет, что действительно может быть сезонная средняя реверсия, происходящая во внутридневное время для случая акций.

Правила стратегии:

1. Нужно выбрать в момент открытия рынка все акции, для которых справедливо следующее: разница между минимумом предыдущего дня и ценой открытия этого дня меньше, чем их стандартное отклонение. Где стандартное отклонение рассчитывается с использованием ежедневных результатов закрытия за последние 90 дней. Это акции, которые “провалились вниз”.
2. Следует сузить этот список акций, требуя, чтобы цены их открытия были выше 20-дневной скользящей средней цен закрытия.
3. Купить 10 акций в этом списке, которые имеют самые низкие доходы от их минимума предыдущего дня. Если в списке меньше 10 акций, то покупайте весь список.
4. Ликвидировать все позиции при закрытии рынка.

Обоснование этой стратегии заключается в том, что в дни, когда фьючерсы на фондовый индекс занижены (просели) перед открытием, некоторые акции недооценены из-за больших продаж перед открытием. Но как только эта паническая продажа закончится, акции постепенно в течение дня восстанавливают позиции.

Правило 2 часто очень полезно в стратегиях средней реверсии. На самом деле это импульсный фильтр, наложенный на стратегию возврата к среднему значению. Обычно те акции, которые упали "чуть-чуть" имеют больше шансов на разворот, чем те, которые упали "много", потому что последние часто имеют негативные новости, такие как плохие объявления о доходах. Падение, вызванное негативными новостями, с меньшей вероятностью восстановится. (Подробнее о стратегиях на основе импульса позже).

Кроме того, тот факт, что цена акции выше, чем долгосрочная скользящая средняя привлекает давление со стороны крупных игроков, таких как долгосрочные фонды, чьи торговые горизонты, как правило, длиннее. Этот спрос на ликвидность на открытии может преувеличить нисходящее давление на цену, но движения цены из-за ликвидности более склонны к возвращению, когда такие требования исчезают, цена движется в фундаментальном направлении. Поэтому данная стратегия может увенчаться успехом в условиях информационного давления, когда традиционная парная внутридневная торговля, скорее всего, не удастся. Рассмотрим пример в части Практика.

Арбитраж между ETF и компонентами (акциями).

Существует известная концепция индексного арбитража, согласно которой трейдер торгует на разнице в стоимости между портфелем акций, составляющим индекс и фьючерсом на этот индекс. Если веса акций правильно распределены, то эти индексы коинтегрированны. Но в силу большой известности этой стратегии как правило сложно извлечь из неё выгоду, рассмотрим концепцию индексного арбитража для случая портфеля акций образующих ETF и сам ETF. В этом случае, выбирается подмножество акций, составляющих ETF, для формирования портфеля. Один из методов составления портфеля – собирать те акции, которые индивидуально коинтегрированны с ETF. Продемонстрируем такой подход на примере известного ETF – SPY. Мы выберем данные за один (в нашем примере, с 1 января 2007 года по 31 декабря 2007 года) в качестве обучающей выборки и будем искать все акции, которые объединяются со SPY с вероятностью не менее 90 процентов, используя тест Йохансена.

Затем мы сформируем портфель этих акций с равным капиталом для каждой акции, и ещё раз проведём тест Йохансена, на предмет проверки того, что этот портфель коинтегрирован с SPY. Этот шаг необходим, потому что произвольное присвоение равного веса капитала каждой акции не обязательно приводит к коинтергации серии цен портфеля акций с ETF, даже если каждая из акций коинтегрированна с SPY. Будем использовать логарифм цены во втором тесте, поскольку ожидаем, что придётся ребалансировать этот портфель каждый день, так чтобы капитал на каждую акцию был постоянен (это свойство логарифмов цен).

Подтверждение части утверждений из этой части можно обнаружить в статье “ Statistical arbitrage in the US equities market”[14]

После подтверждения коинтеграции, мы можем протестировать линейную стратегию средней реверсии, описанную ранее. Реализацию на Python я представлю в главе Практика.

Перекрёстная средняя реверсия: Линейная Long – Short модель.

Трейдинг по стратегии средней реверсии базируется на основе коинтеграции. Формируется портфель с фиксированным набором инструментов и с фиксированным количеством акций или фиксированным капиталом по каждому инструменту. Это фиксированное число может быть определено по желанию, линейной регрессией, тестом Йохансена или ограниченной оптимизацией. Но нет никаких причин считать, что портфель должен состоять из одного и того же набора инструментов с одинаковыми весами каждый день. Для многих портфельных биржевых стратегий исходит зависит именно от разумного ежедневного выбора и балансировки весов акций.

Сейчас мы рассмотрим, так называемую, перекрёстную стратегия средней реверсии (чаще всего включает именно акции, а не фьючерсы и валюту), причём не обязательно, чтобы цена каждой составляющей портфель акции вернулась к своему среднему значению. Основное внимание уделяется их краткосрочной относительной доходности, и мы полагаемся на последовательную анти - корреляцию этих относительных доходов для получения прибыли. В большинстве случаев относительная доходность рассчитывается как доходность акций минус средняя доходность всех акций в определенной выборке. Ключевой чертой стратегии является ожидание того, что в среднем за “неудачным” днём следует “удачный” день и наоборот. Поскольку мы измеряем только относительную доходность, вполне возможно, что мы закроем позицию по акции, даже если предыдущая

[14] Avellaneda, M., & Lee, J. H. (2010). Statistical arbitrage in the US equities market. *Quantitative Finance*, *10*(7), 761-782. https://doi.org/10.1080/14697680903124632

(абсолютная) доходность отрицательна (пока она не так отрицательна, как средняя доходность по всем акциям из выборки).

Одной из интересных особенностей перекрёстной стратегий является то, что, в отличие от «стратегии временных рядов”, мы не должны ожидать прибыли от каждой отдельной акции, так как некоторые из них могут служить “хеджами” на несколько дней. Скорее, прибыль можно получить только в совокупности по всем акциям.

Согласно стратегии, можно начать, с инвестиции не обязательно в равном количестве во все акции индекса (например S&P 500). Затем по окончании каждого торгового дня мы будем назначать новый размер капитала ω_iдля i-ой акции по следующей формуле:

$$\omega_i = -\frac{r_i - \langle r_j \rangle}{\sum_k |r_k - \langle r_j \rangle|}, \tag{12}$$

где r_i – ежедневная доходность i-ой акции, $\langle r_j \rangle$ - ежедневный средний доход по всем акциям индекса.

Другими словами, если акция имеет положительный доход по отношению к остальным акциям мы её продадим, и если она имеет отрицательный доход по отношению к остальным акциям, то мы её покупаем. Следует заметить, что веса нормированы.

Конечно, к этому подходу стоит относиться как к возможному плану действий, разумеется, что он не при любых обстоятельствах выигрышен, например, есть исследование, доказывающее справедливость иного подхода, которых исходит от альтернативных рассуждений: рассматривается ситуация, в которой наличие настроений на рынке сочетается с аргументом о том, что завышение цен должно быть более распространенным, чем занижение цен, из-за препятствий для краткосрочной продажи.[15] Поэтому всегда стоит помнить о возможных альтернативах.

Продемонстрирую стратегию: линейная Long – Short модель на примере в практической главе.

Меж дневные импульсные стратегии.

Существуют несколько основных причин импульса:

1. Медленное распространение, анализ и принятие новой информации.
2. Принудительная продажа или покупка активов различных видов ценных бумаг.
3. Манипуляции рынком высокочастотными трейдерами.

[15] Stambaugh, R. F., Yu, J., & Yuan, Y. (2012). The short of it: Investor sentiment and anomalies. *Journal of Financial Economics, 104*(2), 288-302. https://doi.org/10.1016/j.jfineco.2011.12.001

Мы будем обсуждать торговые стратегии, которые используют каждую причину. Импульс временного ряда очень прост и интуитивно понятен: прошлые доходности временного ряда положительно коррелируют с будущим.

Существует ряд научных публикаций по теме импульсной торговли, в частности, известная статья “Time series momentum”[16], в которой рассматривается прибыль инвестиций в течение одного-двенадцати месяцев частично разворачивается на более длительные горизонты, что в свою очередь согласуется с теориями первоначальной недостаточной реакции и затянувшейся избыточной реакции на настроения. Диверсифицированный портфель стратегий импульса движения временных рядов по всем классам активов обеспечивает значительную аномальную доходность при незначительном воздействии стандартных факторов ценообразования на активы и показывает наилучшие результаты на экстремальных рынках. Изучая торговую деятельность спекулянтов и хеджеров, авторы обнаружили, что спекулянты получают прибыль от импульса временного ряда за счет хеджеров.

Тестирование временного ряда на импульс.

Прежде всего надо выяснить, как заметить и измерить импульс.

Как уже стало ясно, импульс в временном ряду означает, что прошлая доходность положительно коррелирует с будущими доходами. Отсюда следует, что мы можем просто вычислить коэффициент корреляции доходности вместе с ее p-значением (p-value, которая представляет собой вероятность нулевой гипотезы об отсутствии корреляции). В данном случае одной из особенностей расчета коэффициента корреляции является то, что мы должны выбрать конкретный временной интервал, в котором рассматриваем доходность, причём от выбора этих интервалов очень многое зависит, иногда просто поменяв длину интервала, можно изменить результаты.

В результате мы должны найти оптимальную пару прошлого и будущего периодов, которая даст наибольшую положительную корреляцию и использовать прошлый период в качестве обучающей выборки (look-back) и будущий период в качестве периода вложения для нашей стратегии импульса.

---

[16] Moskowitz, T. J., Ooi, Y. H., & Pedersen, L. H. (2012). Time series momentum. *Journal of financial economics, 104*(2), 228-250. https://doi.org/10.1016/j.jfineco.2011.11.003

В качестве альтернативы мы также можем проверить корреляцию между знаками прошлой и будущей доходностью. Это уместно, когда все, что мы хотим знать, это: движение вверх будет сопровождаться еще одним движением вверх, и нас не волнует величины изменений.

В случае заинтересованности в том, есть ли долгосрочные тенденции поведения тренда во временном ряду безотносительно временных интервалов, то мы можем вычислить экспоненту Хёрста (Hurst exponent) вместе с тестом дисперсионного отношения (Variance Ratio test), чтобы исключить нулевую гипотезу случайного блуждания цен.

Далее я проиллюстрирую использование этих тестов на примере двухлетнего казначейского векселя, обращающегося на Чикагской товарно-сырьевой бирже (CME) в качестве примера. Коэффициент корреляции и его p – значение, коэффициент Хёрста и коэффициент вариации можно рассчитать с помощью Python.

При вычислении корреляций пар доходностей в различные периоды времени, мы должны позаботиться о том, чтобы данные были сопоставимы. Если период анализа больше, чем период вложения, мы должны сдвинуться вперед на период вложения, чтобы сгенерировать новую пару доходностей. Если период вложения больше, чем период ожидания, то следует сдвинуться вперёд на период вложения. Пример приведён в разделе Практика.

Пересечение скользящей средней (The moving average crossover).

Данный тип стратегий – это отдельный тип, который не относится не к типу импульсных стратегий, не к обратной реверсии. Стратегия, которую мы здесь рассмотрим имеет следующую основу: мы создаём две отдельные простые скользящие средние (SMA) временного ряда с разными периодами просмотра, например, 40 дней и 100 дней. Если короткая скользящая средняя превышает длинную скользящую среднюю, то переходим в длинную позицию, если длинная скользящая средняя превышает короткую скользящую среднюю, то переходим в короткую позицию. (Важно всегда держать в голове, что когда вы открываете длинную позицию, вы думаете, что цена акций будет расти и в будущем будет продаваться по более высокой цене (= сигнал на покупку); Когда вы открываете короткую позицию, вы продаете свои акции, ожидая, что вы сможете выкупить их по более низкой цене и получить прибыль (= сигнал на продажу).)

Инструкция стратегии шаг за шагом:

1. Мы должны определить два «окна», то есть периода, по которым вычисляется скользящая средняя, это не обязательно 40 и 100 дней, главное, чтобы не

произошло путаницы в коде, что короткое окно соответствует именно меньшему числу.

2. Теперь создайём пустой signals DataFrame (таблицу данных), копируем индекс наших данных (В данном случае – это акции Apple, то есть индекс aapl), чтобы мы могли начать вычислять ежедневный сигнал покупки или продажи для наших aapl данных.
3. Создаём столбец в нашем пустом signals DataFrame с именем signal и инициализируем его, установив значение для всех строк в этом столбце 0.0
4. После подготовительной работы нужно создать набор коротких и длинных простых скользящих средних для соответствующих длинных и коротких окон. Используем rolling() функцию, чтобы начать вычисления скользящего окна: в функции указываем window и min period(), и устанавливаем center аргумент. На практике это приведет к rolling()функции, которой вы передали или, short_window, или long_window, 1 как минимальное количество наблюдений в окне, которые должны иметь значение, и False, таким образом, метки не будут установлены в центре окна. Далее, не забудьте также связать mean()функцию в цепочку, чтобы вычислить скользящее среднее.
5. После того, как мы вычислили среднее для коротких и длинных окон, мы должны создать сигнал для ситуации, когда короткая скользящая средняя пересекает длинную скользящую среднюю, но только в течение периода, превышающего окно самой короткой скользящей средней. В Python, это приведет к условию: signals['short_mavg'][short_window:] > signals['long_mavg'][short_window:]. Важно, что мы добавляем, [short_window:]чтобы выполнить условие «только для периода, превышающего окно с кратчайшим скользящим средним». Когда условие истинно, инициализированное значение 0.0 в signal столбце будет перезаписано 1.0. «Сигнал» создан! Если условие ложно, исходное значение 0.0 будет сохранено и сигнал не генерируется. Мы используете where() функцию NumPy для установки этого условия. Так же, переменная, которой мы присваиваем этот результат signals['signal'][short_window] делается для того, чтобы создавать сигналы только для периода, превышающего окно с кратчайшим скользящим средним.

6. Наконец, мы берём разницу сигналов для генерации реальных торговых приказов. Другими словами, в столбце signals DataFrame мы сможем различать длинные и короткие позиции, то есть покупаем или продаём акции.

Как обычно реализация стратегии на Питоне приведена в главе с практикой.

Глава 2. Практическая реализация автоматических торговых стратегий с использованием программных продуктов.

Замечание: В дальнейшем реализованные программы являются кодами,а не текстом, поэтому я оформляю их иначе для их репрезентативности и удобочитаемости.

Основы для ряда примеров реализации тестирований и стратегий на практике были мною позаимствованы из Chan, E. (2013)[17], а также из Chan, E. (2009)[18].

Демонстрация расширенного теста Дики - Фуллера (ADFtest).

Для примера были взяты данные по валютной паре доллар США к Канадскому доллару в период с 22 июля 2007 года по 28 марта 2012 года.

ADF test как функция в Pythonвоспроизводится в виде adffunction. Имеет три параметра на входе:

Первое, временной ряд в хронологическом порядке.

Второе, параметр, установить значение μ и должен ли дрейф β в уравнении быть равен нулю.

$$\Delta y(t) = \lambda y(t-1) + \mu + \beta_t + \alpha_1 \Delta y(t-1) + \ldots + \alpha_k \Delta y(t-k) + \epsilon_t$$

Мы предполагаем, что значение μ не равно нулю, так как средняя цена, к которой цены возвращаются редко равна нулю. Однако мы должны предположить, что дрейф равен нулю, потому что постоянный дрейф цены имеет тенденцию быть гораздо меньшей величиной, чем ежедневные колебания цен. Эти соображения означают, что второй параметр β_tдолжен быть.

Третье, лаг k. Как правило если подставлять значение 0, то не получится отклонить нулевую гипотезу, однако при k = 1, часто становится возможным отклонение нулевой гипотезы.

Из графика можно заметить, что цены не сильно стационарны

Получаем, что статистика теста ADF примерно -1.84, но критическое значение на уровне 90 процентов -2.594, поэтому нельзя отвергать нулевую гипотезу, что λ равно нулю.

---

[17] Chan, E. (2013). *Algorithmic trading: winning strategies and their rationale* (Vol. 625). John Wiley & Sons. https://doi.org/10.1002/9781118676998

[18] Chan, E. (2009). *Quantitative trading: how to build your own algorithmic trading business* (Vol. 430). John Wiley & Sons. https://doi.org/10.1002/9781119203377

Другими словами, мы не можем считать, что валютная пара находится в неподвижном состоянии, что, возможно, не удивительно, учитывая, что канадский доллар известен как товарная валюта, в то время как доллар США таковой не является. Но заметим, что λ отрицательно, что указывает, что временной ряд, по крайней мере, не в тренде.

Код программы, выполненной на Python:

```
# Example A: Using ADF Test for Mean Reversion
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.stattools import adfuller
from genhurst import genhurst

df=pd.read_csv("inputData_USDCAD.csv", encoding='utf-8')
y=df.loc[df['Time']==1659, 'Close']
plt.plot(y)
plt.xlabel('July 22, 2007, to March 28, 2012')
plt.ylabel('USD.CAD')
plt.title('USD.CAD Price Series')
plt.show()

results=adfuller(y, maxlag=1, regression='c', autolag=None)
print(results)

# Find Hurst exponent
H, pVal=genhurst(np.log(y))
print("H=%fpValue=%f" % (H, pVal))
```

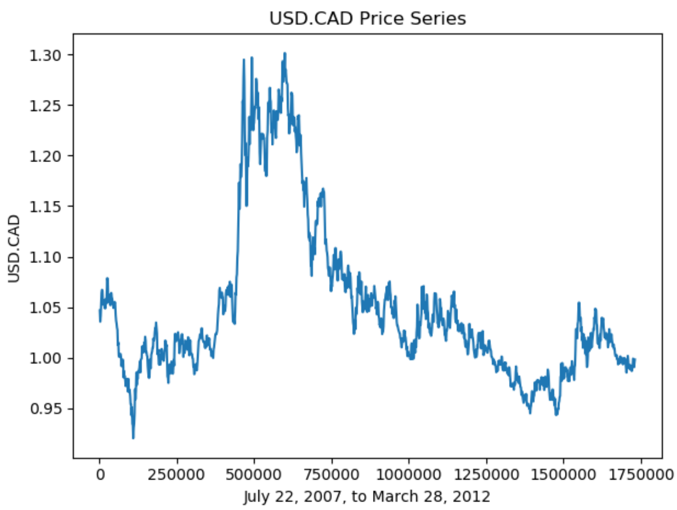


Рис 1. Коинтегрированный временной ряд канадского и американского доллара.

Полученный из программы результат:

(-1.8430182830405577, 0.359322985988917, 1, 1214, {'1%': -3.4357480073434905, '5%': -2.863923702481129, '10%': -2.568039121778048})

H=0.475844 pValue=0.000000

Пример использования теста дисперсионного отношения для проверки стационарности.

Воспользуемся эконометрическим инструментарием MATLAB или Pythonдля применения теста к временном ряду валютной парыUSD.CAD. Если h = 1, то можно отклонить гипотезу случайного блуждания при 90-процентном уровне доверия. Если уровень, h = 0, то это может быть случайное блуждание. pValue – это вероятность того, что нулевая гипотеза (случайные блуждания) верна.

[h,pValue]=vratiotest(log(y));

Получаем, что h = 0 и pValue = 0.367281 для USD.CAD, значит, есть 37-процентный шанс, что это случайное блуждание, поэтому нам нельзя отвергать эту гипотезу.

Расчёт периода возврата (half - life) к среднему значению.

В предыдущем примере мы пришли к выводу, что временной ряд USD.CAD не является стационарным с вероятностью не менее 90 процентов. Но это не обязательно означает, что мы должны отказаться от торговли этим активом с использованием модели среднего реверсирования (meanreversion), потому что большинство стратегий не требуют такого высокого уровня определенности.

Теперь определять, будет ли USD.CAD хорошим кандидатом для торговли со средней реверсией, определим через период возврата к среднему значению при средней реверсии.

Для определения λ в уравнениях (0) и (5) можно провести регрессию $y(t) - y(t - 1)$ в качестве зависимой переменной и $y(t - 1)$ в качестве независимой переменной. Функция метода наименьших квадратов (ols), а также функция лага можно использовать в пакете jplv7.

Код реализован на языке Matlab:

```
load('inputData_USDCAD', 'tday', 'hhmm', 'cl');
y=cl(hhmm==1659);
% Find value of lambda and thus the halflife of mean reversion by linear regression fit
ylag=lag(y, 1);  % lag is a function in the jplv7 (spatial-econometrics.com) package.
deltaY=y-ylag;
deltaY(1)=[]; % Regression functions cannot handle the NaN in the first bar of the time series.
ylag(1)=[];
regress_results=ols(deltaY, [ylag ones(size(ylag))]); % ols is a function in the jplv7 (spatial-econometrics.com) package.
halflife=-log(2)/regress_results.beta(1);
fprintf(1, 'halflife=%f days\n', halflife);
% halflife=115.209794 days
```

В итоге период возврата (полураспада) составляет немного более 115 дней. Для разных ситуаций это может иметь разное значение, может быть долго, может – нет. В любом случае теперь есть понимание того, какой период времени требуется для возвращения к среднему значению.

Целая программа по материалу:

- ADF test
- Поиск показателя Хёрста
- Поиск продолжительности возврата (период полураспада)
- Применение простой линейной стратегии возврата к среднему значению

```
clear;
```

```
% 1 minute data on USDCAD
load('inputData_USDCAD', 'tday', 'hhmm', 'cl');
% Select the daily close at 16:59 ET.
y=cl(hhmm==1659);
plot(y);
% Assume a non-zero offset but no drift, with lag=1.
results=adf(y, 0, 1); % adf is a function in the jplv7 (spatial-econometrics.com) package.
% Print out results
prt(results);
% Augmented DF test for unit root variable:                variable   1
%  ADF t-statistic       # of lags   AR(1) estimate
%       -1.840744              1        0.994120
%
%    1% Crit Value    5% Crit Value   10% Crit Value
%          -3.458          -2.871          -2.594
% Find Hurst exponent
H=genhurst(log(y), 2);
fprintf(1, 'H2=%f\n', H);
% Variance ratio test from Matlab Econometrics Toolbox
[h,pValue]=vratiotest(log(y));
fprintf(1, 'h=%i\n', h); % h=1 means rejection of random walk hypothesis, 0 means it is a random walk.
fprintf(1, 'pValue=%f\n', pValue); % pValue is essentially the probability that the null hypothesis (random walk) is true.
% Output:
% h=0
% pValue=0.367281
% Find value of lambda and thus the halflife of mean reversion by linear regression fit
ylag=lag(y, 1);  % lag is a function in the jplv7 (spatial-econometrics.com) package.
deltaY=y-ylag;
deltaY(1)=[]; % Regression functions cannot handle the NaN in the first bar of the time series.
ylag(1)=[];
regress_results=ols(deltaY, [ylag ones(size(ylag))]); % ols is a function in the jplv7 (spatial-econometrics.com) package.
halflife=-log(2)/regress_results.beta(1);
fprintf(1, 'halflife=%f days\n', halflife);
% halflife=115.209794 days

% Apply a simple linear mean reversion strategy to USDCAD
lookback=round(halflife); % setting lookback to the halflife found above
mktVal=-(y-movingAvg(y, lookback))./movingStd(y, lookback);
```

```
% capital in number of shares invested in USDCAD. movingAvg and movingStd are %
functions from epchan.com/book2
pnl=lag(mktVal, 1).*(y-lag(y, 1))./lag(y, 1); % daily P&L of the strategy
pnl(isnan(pnl))=0;
figure;
plot(cumsum(pnl)); % Cumulative P&L
```

Использование CADF теста для коинтеграции.

Индексные фонды (ETF)дают благодатную почву для поиска сопоставимых по движению временных рядов. В результате они являются хорошими кандидаты для парной торговли. Например, канадская и австралийская экономики основаны на сырьевых товарах, поэтому они по всей видимости, коинтегрированы (буду использовать именно это слово, поскольку в англоязычной литературе используется слово “cointegrate” под этим термином понимается совмещение). Заранее подготовленный временной ряд EWA содержится в массиве x, а EWC содержится в массиве y.Из рисунка 2 можно увидеть, что они выглядят похоже. Точечный график, гдеEWAпо оси абсцисс и EWC по оси ординат, я привёл на рисунке 3 убедительнее демонстрирует совпадение, так как ценовые пары расположены по прямой.

Для оптимального хеджирования (поиска оптимального коэффициента хеджирования (hedgeRatio)) в Python используется функция ols(), взятой из библиотеки statsmodels.formula.api. Как можно заметить из рисунка 4. Временной ряд выглядит довольно стационарным. Мы используем функцию coint() пакета statsmodels.tsa.statools для составления CADFтеста. Результатом функции для входных данных будут три ответа: t - статистика, вероятность (pvalue) и критические значения. Как и для ADFтеста работают предположения, что может быть ненулевое смещение временного ряда портфельной пары, но дрейф равен нулю.

Отмечу, что как в регрессии, так и в тесте CADF мы выбрали EWA быть независимой переменной x, а EWC – зависимой переменной y. Важным моментом является то, что, если мы поменяем роли EWA и EWC, то результат CADFтеста изменится. Коэффициент хеджирования (hedgeRatio), полученный из выборки EWC в качестве независимой переменной не будет точной обратной величиной тому коэффициенту, который получен из выбора EWA в качестве независимой переменной. Во многих случаях (хотя и не для пар EWAEWC, как мы подтвердим позже с помощью теста Йохансена), только один из коэффициентов хеджирования является “правильными” в том смысле, что только один коэффициент хеджирования будет приводить к стационарному портфелю.

Иными словами, при использовании CADFтеста следует попробовать каждую переменную как независимую и посмотреть, какое значение принимает лучшая (наиболее отрицательная) t-статистика и использовать это значение для получения коэффициента хеджирования. В нашем случае полагаем EWAзависимой переменной и запускаем тест CADF. Выясняется, что вADFтестеt-статистика принимает значение -3.06, что конечно, более отрицательное значение, чем критическое значение на 90-процентном уровне доверия -3.047. Поэтому мы можем отвергнуть нулевую гипотезу о том, что λ равно нулю. Другими словами, EWA и EWC коинтегрированны с 90-процентной уверенностью.

Код программы, реализованный на языке Matlab:

```
clear;

% 1 minute data on EWA-EWC
load('inputData_ETF', 'tday', 'syms', 'cl');
idxA=find(strcmp('EWA', syms));
idxC=find(strcmp('EWC', syms));
x=cl(:,idxA);
y=cl(:,idxC);
plot(x);
hold on;
plot(y, 'g');

legend('EWA', 'EWC');
xlabel('April4, 2006, to April 9, 2012');
ylabel('Share price $');
figure;
scatter(x, y);
xlabel('EWA share price');
ylabel('EWC share price');
figure;
regression_result=ols(y, [x ones(size(x))]);
hedgeRatio=regression_result(1);
```

```
plot(y-hedgeRatio*x);
xlabel('April 26, 2006, to April 9, 2012');
ylabel('EWC - hedgeRatio*EWA');
% Assume a non-zero offset but no drift, with lag=1.
import cadf from jplv7
results=cadf(y, x, 0, 1); % cadf is a function in the jplv7 (spatial-econometrics.com) package. We pick y to be the dependent variable again.
% Print out results
prt(results);

% Output:
%  Augmented DF test for co-integration variables:      variable   1,variable   2
% CADF t-statistic        # of lags   AR(1) estimate
%      -3.64346635               1        -0.020411
%
%    1% Crit Value    5% Crit Value   10% Crit Value
%          -3.880           -3.359           -3.038
% Combine the two time series into a matrix y2 for input into Johansen test
y2=[y, x];
results=johansen(y2, 0, 1); % johansen test with non-zero offset but zero drift, and with the lag k=1.
% Print out results
prt(results);
% Output:
%  Johansen MLE estimates
% NULL:                  Trace Statistic       Crit 90%       Crit 95%       Crit 99%
% r <= 0   variable   1          19.983          13.429          15.494          19.935
% r <= 1   variable   2           3.983           2.705           3.841           6.635
%
% NULL:                  Eigen Statistic       Crit 90%       Crit 95%       Crit 99%
% r <= 0   variable   1          16.000          12.297          14.264          18.520
```

```
% r <= 1   variable   2          3.983          2.705          3.841          6.635
% Adding IGE to the portfolio
idxI=find(strcmp('IGE', syms));
z=cl(:,idxI);
y3=[y2, z];
results=johansen(y3, 0, 1); % johansen test with non-zero offset but zero drift, and with the lag k=1.
% Print out results
prt(results);
% Output:
%  Johansen MLE estimates
% NULL:                 Trace Statistic       Crit 90%       Crit 95%       Crit 99%
% r <= 0   variable   1          34.429          27.067          29.796          35.463
% r <= 1   variable   2          17.532          13.429          15.494          19.935
% r <= 2   variable   3          4.471          2.705          3.841          6.635
%
% NULL:                 Eigen Statistic       Crit 90%       Crit 95%       Crit 99%
% r <= 0   variable   1          16.897          18.893          21.131          25.865
% r <= 1   variable   2          13.061          12.297          14.264          18.520
% r <= 2   variable   3          4.471          2.705          3.841          6.635
results.eig % Display the eigenvalues
% ans =
%
%    0.0112
%    0.0087
%    0.0030
results.evec % Display the eigenvectors
% ans =
%
%   -1.0460   -0.5797   -0.2647
%    0.7600   -0.1120   -0.0790
```

```
%     0.2233    0.5316    0.0952

yport=sum(repmat(results.evec(:, 1)', [size(y3, 1) 1]).*y3, 2); % (net) market value of portfolio
% Find value of lambda and thus the halflife of mean reversion by linear regression fit
ylag=lag(yport, 1);  % lag is a function in the jplv7 (spatial-econometrics.com) package.
deltaY=yport-ylag;
deltaY(1)=[]; % Regression functions cannot handle the NaN in the first bar of the time series.
ylag(1)=[];
regress_results=ols(deltaY, [ylag ones(size(ylag))]); % ols is a function in the jplv7 (spatial-econometrics.com) package.
halflife=-log(2)/regress_results.beta(1);
fprintf(1, 'halflife=%f days\n', halflife);
% halflife=22.662578  days
%
%  Apply a simple linear mean reversion strategy to EWA-EWC-IGE
lookback=round(halflife); % setting lookback to the halflife found above
numUnits =-(yport-movingAvg(yport, lookback))./movingStd(yport, lookback); % capital invested in portfolio in dollars.  movingAvg and movingStd are functions from epchan.com/book2
positions=repmat(numUnits, [1 size(y3, 2)]).*repmat(results.evec(:, 1)', [size(y3, 1) 1]).*y3; % results.evec(:, 1)' can be viewed as the capital allocation, while positions is the dollar capital in each ETF.
pnl=sum(lag(positions, 1).*(y3-lag(y3, 1))./lag(y3, 1), 2); % daily P&L of the strategy
ret=pnl./sum(abs(lag(positions, 1)), 2); % return is P&L divided by gross market value of portfolio
ret(isnan(ret))=0;
figure;
plot(cumprod(1+ret)-1); % Cumulative compounded return
fprintf(1, 'APR=%f Sharpe=%f\n', prod(1+ret).^(252/length(ret))-1, sqrt(252)*mean(ret)/std(ret));
% APR=0.125739 Sharpe=1.391310
```

Реализация на Python:

```
# Using the CADF test for cointegration
import numpy as np
import pandas as pd
import matplotlib.pyplotas plt
import statsmodels.formula.apias sm
import statsmodels.tsa.stattoolsas ts
import statsmodels.tsa.vector_ar.vecmas vm
df=pd.read_csv('inputData_EWA_EWC_IGE.csv')
df['Date']=pd.to_datetime(df['Date'], format='%Y%m%d').dt.date#
remove HH:MM:SS
df.set_index('Date', inplace=True)
df.plot()
df.plot.scatter(x='EWA', y='EWC')
plt.xlabel('EWA share price')
plt.ylabel('EWC share price')
plt.xlabel('April 4,2006, to April 9,2012')
plt.ylabel('Share price $')
plt.show()
results=sm.ols(formula="EWC ~ EWA", data=df[['EWA',
'EWC']]).fit()
print(results.params)
hedgeRatio=results.params[1]
print('hedgeRatio=%f' % hedgeRatio)
(df['EWC']-hedgeRatio*df['EWA']).plot()
plt.xlabel('Stationarity of Residuals of Linear ')
plt.ylabel('EWC - hedgeRatio*EWA')
plt.show()
print(ts.coint(df['EWA'], df['EWC']))
# cadf test
coint_t, pvalue, crit_value=ts.coint(df['EWA'], df['EWC'])
print('t-statistic=%f' % coint_t)
print('pvalue=%f' % pvalue)
print(crit_value)
# Johansen test
```

```
result=vm.coint_johansen(df[['EWA', 'EWC']].values, det_order=0,
k_ar_diff=1)
print('Johansen test')
print(result.lr1)
print(result.cvt)
print(result.lr2)
print(result.cvm)
# Add IGE for Johansen test
result=vm.coint_johansen(df.values, det_order=0, k_ar_diff=1)
print('Add IGE for Johansen test')
print(result.lr1)
print(result.cvt)
print(result.lr2)
print(result.cvm)
print('eigenvalues')
print(result.eig)  # eigenvalues
print('eigenvectors')
print(result.evec)  # eigenvectors
yport=pd.DataFrame(np.dot(df.values, result.evec[:, 0])) #
(net) market value of portfolio
ylag=yport.shift()
deltaY=yport-ylag
df2=pd.concat([ylag, deltaY], axis=1)
df2.columns=['ylag', 'deltaY']
regress_results=sm.ols(formula="deltaY ~ ylag", data=df2).fit()
# Note this can deal with NaN in top row
print(regress_results.params)

halflife=-np.log(2)/regress_results.params['ylag']
print('halflife=%f days' % halflife)

#  Apply a simple linear mean reversion strategy to EWA-EWC-IGE
lookback=np.round(halflife).astype(int) #  setting lookback to
the halflife found above
numUnits =-(yport-
```

```
yport.rolling(lookback).mean())/yport.rolling(lookback).std() # capital invested in portfolio in dollars.  movingAvg and movingStd are functions from epchan.com/book2
positions=pd.DataFrame(np.dot(numUnits.values, np.expand_dims(result.evec[:, 0], axis=1).T)*df.values) # results.evec(:, 1)' can be viewed as the capital allocation, while positions is the dollar capital in each ETF.
pnl=np.sum((positions.shift().values)*(df.pct_change().values), axis=1) # daily P&L of the strategy
ret=pnl/np.sum(np.abs(positions.shift()), axis=1)
(np.cumprod(1+ret)-1).plot()
print('APR=%f Sharpe=%f' % (np.prod(1+ret)**(252/len(ret))-1, np.sqrt(252)*np.mean(ret)/np.std(ret)))
# APR=0.125739 Sharpe=1.391310APR = Annual Percent Rate
```

Полученные результаты:

Intercept   6.411331

EWA   0.962429

dtype: float64

hedgeRatio=0.962429

(-3.063528097618725, 0.09586561374353125, array([-3.90376106, -3.34020915, -3.04728056]))

t-statistic=-3.063528

pvalue=0.095866

[-3.90376106 -3.34020915 -3.04728056]

Johansen test

[19.98321869 3.98276124]

[[13.4294 15.4943 19.9349]

 [2.7055 3.8415 6.6349]]

[16.00045745 3.98276124]

[[12.2971 14.2639 18.52]

 [2.7055 3.8415 6.6349]]

Add IGE for Johansen test

[34.42862022 17.53171895 4.47102054]

```
[[27.0669 29.7961 35.4628]
 [13.4294 15.4943 19.9349]
 [ 2.7055  3.8415  6.6349]]
[16.89690127 13.06069841  4.47102054]
[[18.8928 21.1314 25.865 ]
 [12.2971 14.2639 18.52  ]
 [ 2.7055  3.8415  6.6349]]
eigenvalues
[0.01121626 0.00868086 0.00298021]
eigenvectors
[[ 0.7599635  -0.11204898  0.0789828 ]
 [-1.04602749 -0.5796762   0.26467204]
 [ 0.22330592  0.53159644 -0.09515547]]
Intercept   -0.115768
ylag        -0.030586
dtype: float64
halflife = 22.662578 days
APR = 0.125739 Sharpe = 1.402653
```

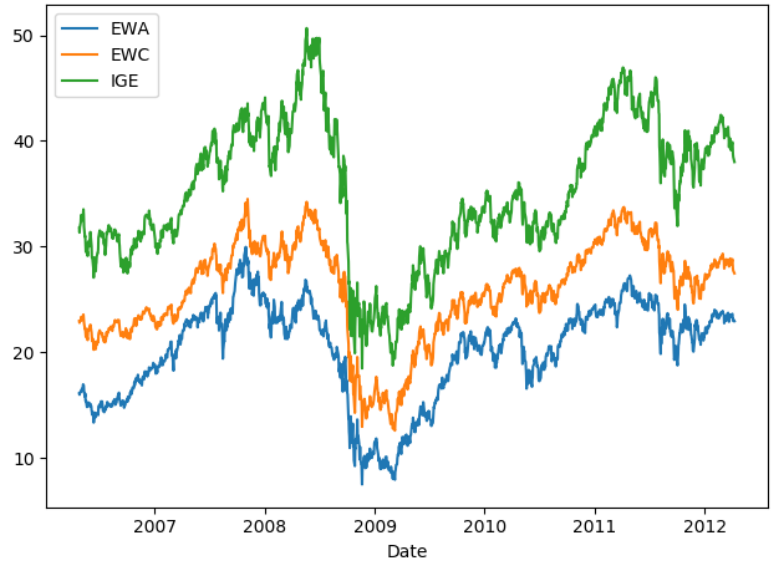


Рис 2. Три коинтегрированных временных ряда.

Использование теста Йохансена для коинтеграции.

Возьмем временной ряд EWA и EWC и применим к нему тест Йохансена. Сделаем это через Python. В библиотеке statsmodels.tsa.vector_ar.vecm возьмём функцию coint_johansen для проведения теста Йохансена. Функция на вход получает три группы данных (Y, p, k):

- Y - Сами временные ряды в виде dataframe (матрица цен, в которой каждый вектор столбец представляет один временной ряд)
- p = 0, (установили через ADFи CADF тесты, в уравнении (7) полагаем среднее значение для реверсии $M \neq 0$, но постоянный член дрейфа $\beta = 0$)
- k - это количество лагов, которое мы снова задаем равным 1.

Полученные значения Johansentest:

[19.98321869 3.98276124]

[[13.4294 15.4943 19.9349]

[2.7055 3.8415 6.6349]]

[16.00045745 3.98276124]

[[12.2971 14.2639 18.52]

[2.7055 3.8415 6.6349]]

Мы видим, что для теста статистики трассировки гипотеза $r = 0$ отклоняется с уверенностью 99%, и $r \leq 1$ отклоняется с вероятностью 95%. По собственной (в алгебраическом смысле собственной) статистике делает вывод, что гипотеза $r = 0$отклоняется с вероятностью 95%, и $r \leq 1$ отклоняется с вероятностью 95%. Это означает, что из обоих тестов мы делаем вывод, что EWA и EWC коинтегрированы.

Принцип поиска статистик:

% NULL:	Trace Statistic	Crit 90%	Crit 95%	Crit 99%
% r <= 0 variable 1	19.983	13.429	15.494	19.935
% r <= 1 variable 2	3.983	2.705	3.841	6.635

%

% NULL:	Eigen Statistic	Crit 90%	Crit 95%	Crit 99%
% r<= 0 variable 1	16.000	12.297	14.264	18.520
% r<= 1 variable 2	3.983	2.705	3.841	6.635

Достоинством теста Йохансена является то, что не нужно производить его дважды, меняя зависимые и независимые переменные, как в CADFтесте. В этом случае, в качестве ответа получаются все независимые коинтегрированных отношений, которые существуют. Другими словами тест Йохансена независим от порядка временных рядов.

Если изменить портфель, добавив в него новый ETF–IGE (ETF, состоящий из котировок цен природных ресурсов). Теперь, обозначим за z– массив, содержащий три временных ряда (EWA,EWC,IGE). Проведем тест Йохансена на всех трёх серий цен разобраться, сколько коинтеграционных отношений мы сможем получить из этого трио.

Полученные значения для этого случая Add IGE for Johansen test:

[34.42862022 17.53171895 4.47102054]

[[27.0669 29.7961 35.4628]

[13.4294 15.4943 19.9349]

[2.7055 3.8415 6.6349]]

[16.89690127 13.06069841 4.47102054]

[[18.8928 21.1314 25.865]

[12.2971 14.2639 18.52]

[2.7055 3.8415 6.6349]]

eigenvalues

[0.01121626 0.00868086 0.00298021]

eigenvectors

[[0.7599635 -0.11204898 0.0789828]

[-1.04602749 -0.5796762 0.26467204]

[0.22330592 0.53159644 -0.09515547]]

Вывод: И статистика трассировки и собственная статистика теста делают вывод, что мы имеем три коинтегрированных отношения с 95-процентной уверенностью.

(как понять)

% NULL:	Trace Statistic	Crit 90%	Crit 95%	Crit 99%
% r <= 0 variable 1	34.429	27.067	29.796	35.463
% r <= 1 variable 2	17.532	13.429	15.494	19.935
% r <= 2 variable 3	4.471	2.705	3.841	6.635

%

% NULL:	Eigen Statistic	Crit 90%	Crit 95%	Crit 99%
% r <= 0 variable 1	16.897	18.893	21.131	25.865
% r <= 1 variable 2	13.061	12.297	14.264	18.520
% r <= 2 variable 3	4.471	2.705	3.841	6.635

Также, следует обратить внимание на порядок собственных значений и собственных векторов на выводе (слева направо расположены собственные вектора по убыванию их собственных значений)

eigenvalues

[0.01121626 0.00868086 0.00298021]

eigenvectors

```
[[ 0.7599635  -0.11204898  0.0789828 ]
 [-1.04602749 -0.5796762   0.26467204]
[ 0.22330592  0.53159644 -0.09515547]]
```

Этот порядок означает, что первое коинтеграционное отношение будет самым “сильным”, потому что имеет кратчайший период возврата к среднему значению. Поэтому имеет смысл выбрать именно этот собственный вектор для формирования нашего стационарного портфеля (собственный вектор определяет доли каждого ETF в общем портфеле), и мы можем найти период возврата этого портфеля таким же методом, как раньше, когда мы имели дело стационарным временным рядом. Единственная разница в том, что теперь мы должны вычислять массив T × 1yport, представляющий чистую рыночную стоимость(цену) портфеля, который равен количеству акций каждого ETF, умноженному на цену акций каждого ETF, затем суммируем по всем ETF. В примере yport берет на себя роль y. В коде приведены расчёты и на выходе имеем период возврата 23 дня. (Если сравнить с парой USD.CADдля которой 115 дней, можно заметить разницу и отдать предпочтение этому трио).

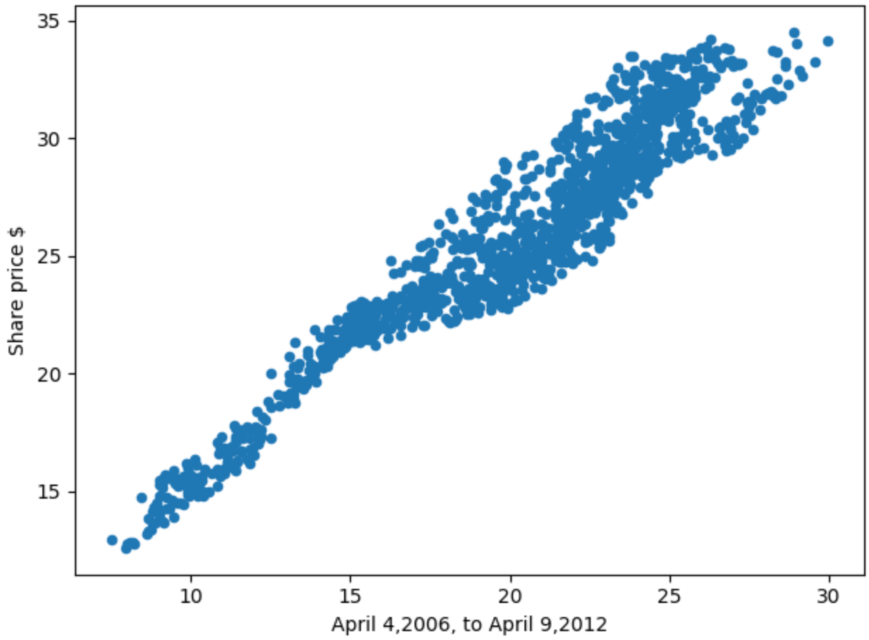


Рис 3.

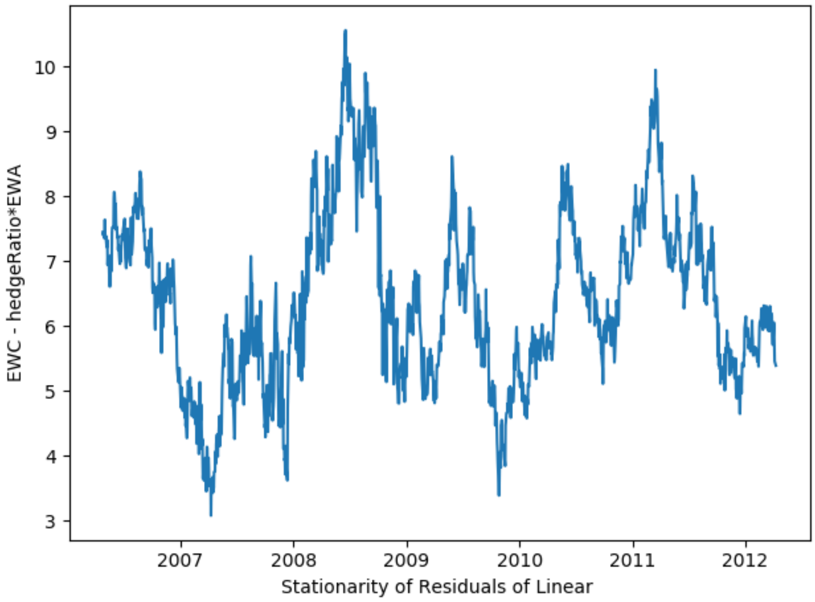


Рис 4.

Таким образом, интерпретируя данные, полученные из теста Йохансена, можно утверждать, что тест Йохансена подтверждает выводы CADF теста о том, что пара коинтегрированна. Но, что более интересно, мы добавили еще один ETFк смеси: IGE (это ETF, состоящий из запасов природных ресурсов). Получим, что много коинтеграций могут быть найдены из этих трёх рядов цен. Мы также используем собственные векторы для формирования стационарного портфеля, и выясняется, что это интерпретируется как период возврата для средней реверсии.

Пример стратегии возвращения к среднему значению на основе ценового спреда, спреда логарифмов цен и коэффициентов.

Некоторые трейдеры считают, что когда цены на нефть растут, растут и цены на золото. Логика в том, что высокие цены на нефть приводят к росту инфляции, а цены на золото положительно коррелируют с инфляцией. Но, проверить с помощью одного из тестов коинтеграции, которые мы рассмотрели ранее, убеждаемся, что золото (представлено ETF GLD) и цены на нефть (представленные USO) не являются коинтегрированными. Тем не менее, мы небольшой промежуток времени и ответим на вопрос: достаточно ли краткосрочной средней реверсии, чтобы сделать стратегию возврата к среднему значению прибыльной.

Сначала мы попробуем спред цены в качестве сигнала. Однако, нам нужно пересчитывать коэффициент хеджирования каждый день, используя короткий период ретроспективного анализа (20 торговых дней) для адаптации к изменяющимся уровням ETF со временем. Коэффициент хеджирования найдём, как и раньше с помощью функции olsв Python. (Держа в уме, что можно и использовать первый собственные вектор из теста Йохансена).

Заранее подготовленные ценовые массивы GLD - X (Tx1) и USO – Y (Tx1).

Отмечу, что обычно под спредом имеем в виду USO – hedgeRatio * GLD, что равно цене единице портфеля, обозначенный через yport.

Код программы (PriceSpread.py):

```
# Trading Price Spread
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.formula.api as sm
import statsmodels.tsa.stattools as ts
import statsmodels.tsa.vector_ar.vecm as vm
df=pd.read_csv('inputData_GLD_USO.csv')
df['Date']=pd.to_datetime(df['Date'],  format='%Y%m%d').dt.date#
remove HH:MM:SS
df.set_index('Date', inplace=True)
lookback=20
hedgeRatio=np.full(df.shape[0], np.nan)
for t in np.arange(lookback, len(hedgeRatio)):
```

```
regress_results=sm.ols(formula="USO ~ GLD", data=df[(t-
lookback):t]).fit() # Note this can deal with NaN in top row
hedgeRatio[t-1]=regress_results.params[1]
yport=np.sum(ts.add_constant(-hedgeRatio)[:, [1,0]]*df, axis=1)
yport.plot()
plt.xlabel('May 24, 2006, to April 9, 2012')
plt.ylabel('Spread = USO-hedgeRatio*GLD')
plt.show()

# Apply a simple linear mean reversion strategy to GLD-USO
numUnits =-(yport-
yport.rolling(lookback).mean())/yport.rolling(lookback).std() #
capital invested in portfolio in dollars.  movingAvg and
movingStd are functions from epchan.com/book2
positions=pd.DataFrame(np.tile(numUnits.values, [2, 1]).T *
ts.add_constant(-hedgeRatio)[:, [1,0]] *df.values) #
results.evec(:, 1)' can be viewed as the capital allocation,
while positions is the dollar capital in each ETF.
pnl=np.sum((positions.shift().values)*(df.pct_change().values),
axis=1) # daily P&L of the strategy
ret=pnl/np.sum(np.abs(positions.shift()), axis=1)
(np.cumprod(1+ret)-1).plot()
print('APR=%f Sharpe=%f' % (np.prod(1+ret)**(252/len(ret))-1,
np.sqrt(252)*np.mean(ret)/np.std(ret))
```

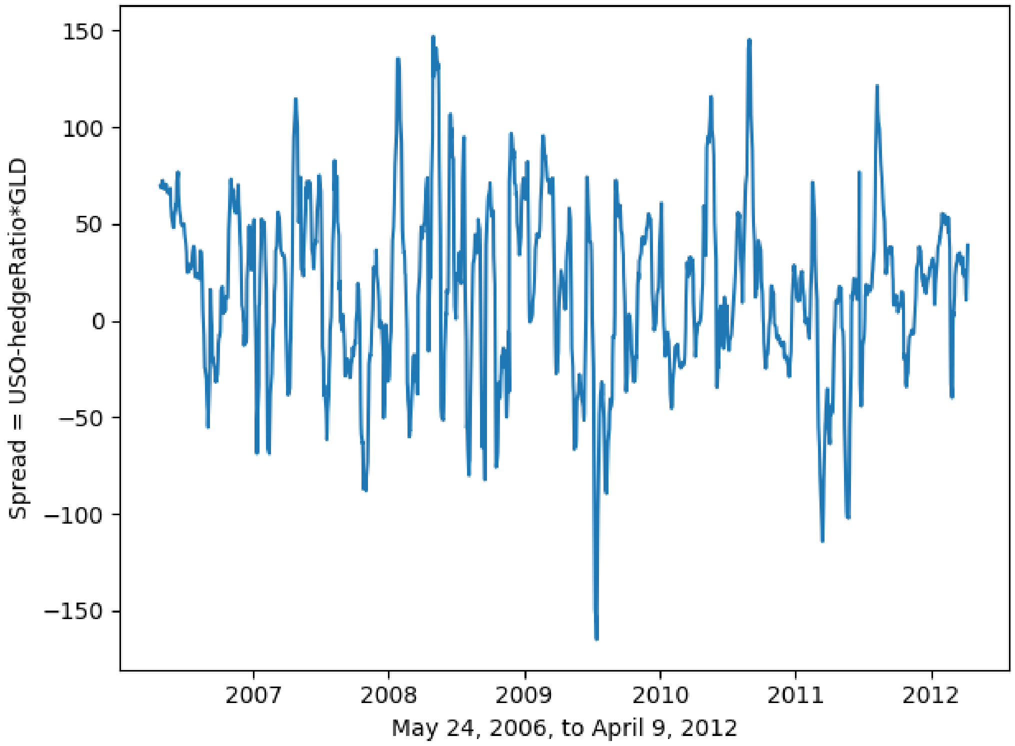


Рис. 5. Спред между USO и GLD с помощью меняющегося коэффициента хеджирования.

Результаты кода:

APR=0.109551 Sharpe=0.598899

“APR = Annual Percentage Rate (годовая процентная ставка), Sharpe = Коэффициент Шарпа”

Рисунок 5 выглядит довольно стационарным.

Теперь посмотрим, сможем ли мы создать линейную стратегию средней реверсии. Ещё раз, количество единиц (акций) блока портфеля который меня создаём должно иметь отрицательный Z-Score, массив позиций Tx2 представляет рыночную стоимость (в долларах) каждой позиции из составляющих ETF, которую мы должны инвестировать.

Мы получаем годовую процентную ставку (APR) около 10,9 процента и коэффициент Шарпа около 0,59 с использованием спреда цены с динамическим коэффициентом хеджирования, даже если GLD и USO никоим образом не коинтегрированы.

Применяя подход средней реверсии к спреду логарифма цен код будет изменён и получаем (LogPriceSpread.py):

```
# Trading Log Price Spread
import numpyas np
import pandas as pd
```

```
import statsmodels.formula.apias sm
import statsmodels.tsa.stattoolsas ts
import matplotlib.pyplotas plt
df=pd.read_csv('inputData_GLD_USO.csv')
df['Date']=pd.to_datetime(df['Date'],  format='%Y%m%d').dt.date# remove
HH:MM:SS
df.set_index('Date', inplace=True)
lookback=20
hedgeRatio = np.full(df.shape[0], np.nan)
for t in np.arange(lookback, len(hedgeRatio)):
regress_results=sm.ols(formula="USO ~ GLD", data=np.log(df[(t-
lookback):t])).fit() # Note this can deal with NaN in top row
hedgeRatio[t-1]=regress_results.params[1]


yport=np.sum(ts.add_constant(-hedgeRatio)[:, [1,0]]*np.log(df), axis=1)
yport.plot()
plt.xlabel('May 24, 2006, to April 9, 2012')
plt.ylabel('Spread = log(USO)-hedgeRatio*log(GLD)')
plt.show()

# Apply a simple linear mean reversion strategy to GLD-USO
numUnits =-(yport-
yport.rolling(lookback).mean())/yport.rolling(lookback).std() # capital
invested in portfolio in dollars.  movingAvg and movingStd are functions from
epchan.com/book2
positions=pd.DataFrame(np.tile(numUnits.values, [2, 1]).T * ts.add_constant(-
hedgeRatio)[:, [1,0]] ) #  positions is the dollar capital in each ETF.
pnl=np.sum((positions.shift().values)*(df.pct_change().values), axis=1) #
daily P&L of the strategy
ret=pnl/np.sum(np.abs(positions.shift()), axis=1)
(np.cumprod(1+ret)-1).plot()
print('APR=%f Sharpe=%f' % (np.prod(1+ret)**(252/len(ret))-1,
np.sqrt(252)*np.mean(ret)/np.std(ret)))
```

результаты кода:

APR=0.099520 Sharpe=0.553442

Таким образом, получили, что стратегия средней реверсии в случае спреда логарифмов цен приносит 9,95% в год с коэффициентом Шарпа 0.553442.

Но на самом деле следует также учесть и в случае спреда цен, и в случае спреда логарифмов цен, что не учтены трансакционные издержки, особенно по ежедневным перебалансировкам портфеля.

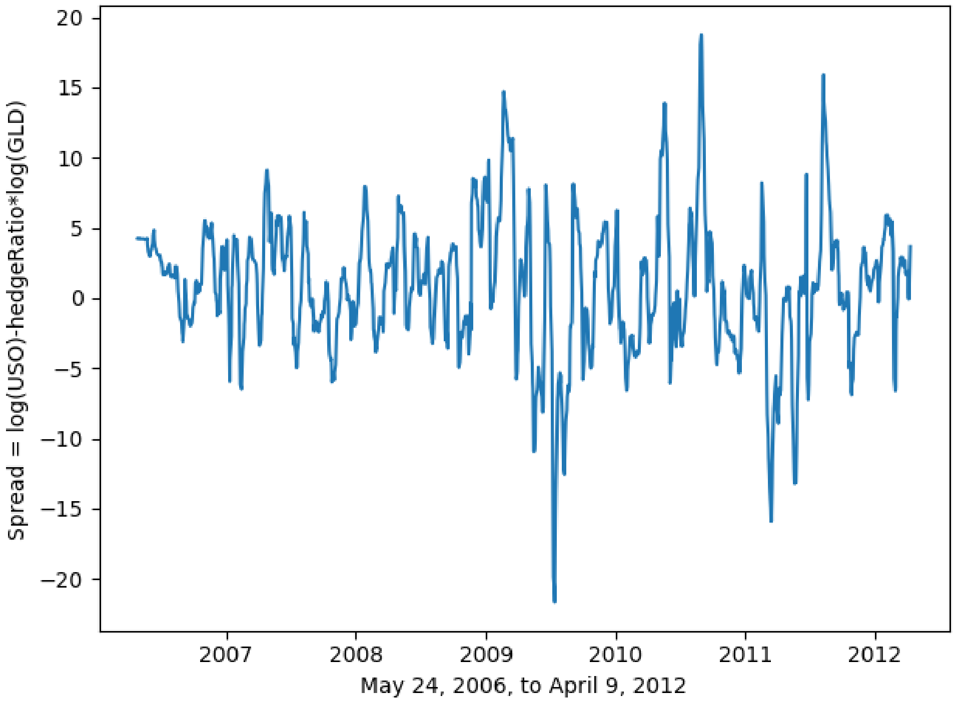


Рис. 6. Спред между логарифмами цен USO и GLD с помощью меняющегося коэффициента хеджирования.

Теперь мы попробуем использовать коэффициент (отношение акций) в качестве сигнала. В этом случае нам также нужно в одном случае продавать, а в другом - покупать, чтобы иметь одинаковый долларовый капитал.

Исходный код программы ниже (Ratio.py):

```
# Trading Ratio
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.tsa.stattools as ts
df=pd.read_csv('inputData_GLD_USO.csv')
df['Date']=pd.to_datetime(df['Date'],  format='%Y%m%d').dt.date#
remove HH:MM:SS
df.set_index('Date', inplace = True)
```

```
lookback=20
ratio=df['USO']/df['GLD']
ratio.plot()
plt.xlabel('May 24, 2006, to April 9, 2012')
plt.ylabel('USO/GLD')
plt.show()
#yport=np.sum(ts.add_constant(-hedgeRatio)[:, [1,0]]*np.log(df), 
axis=1)
#yport.plot()
# Apply a simple linear mean reversion strategy to GLD-USO
numUnits =-(ratio-
ratio.rolling(lookback).mean())/ratio.rolling(lookback).std() # 
capital invested in portfolio in dollars.  movingAvg and 
movingStd are functions from epchan.com/book2
positions=pd.DataFrame(np.tile(numUnits.values, [2, 1]).T * 
np.ones((numUnits.shape[0], 2)) * np.array([-1, 1]) ) # 
positions in dollar invested
pnl=np.sum((positions.shift().values)*(df.pct_change().values), 
axis=1) # daily P&L of the strategy
ret=pnl/np.sum(np.abs(positions.shift()), axis=1)
(np.cumprod(1+ret)-1).plot()
print('APR=%f Sharpe=%f' % (np.prod(1+ret)**(252/len(ret))-1, 
np.sqrt(252)*np.mean(ret)/np.std(ret)))
```

Результаты программы:

APR=-0.140674 Sharpe=-0.749583

Отрицательная доходность (-14,07% годовых, при коэффициенте Шарпа -0,75)

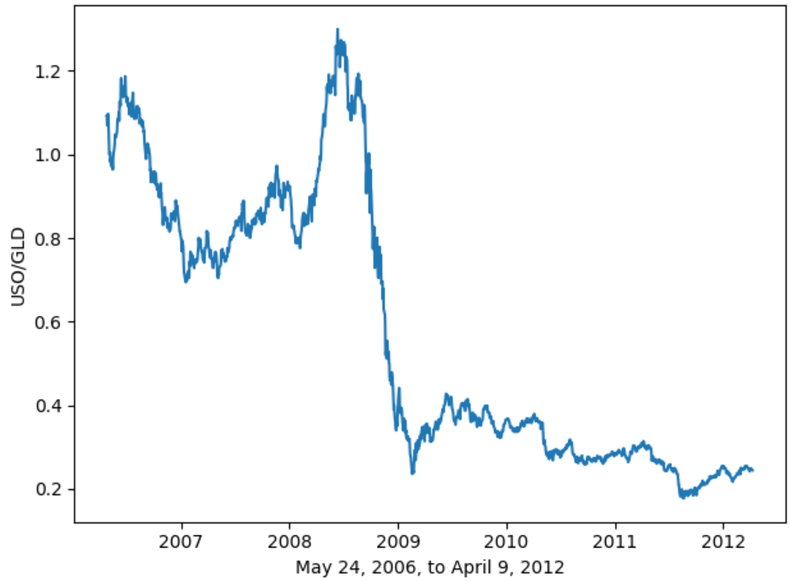


Рис. 7. Отношение между USO и GLD.

Заметим, что соотношение на самом деле не выглядит очень стационарным, по сравнению либо с ценовым спредом, либо с адаптивным коэффициентом хеджирования. Поэтому нас не должен удивлять тот факт, что данная модель стратегии средней реверсии показывает плохие результаты с отрицательным среднегодовым процентом.

Модель Bue – on – Gap.

Для начала понадобятся несколько входных данных:

- Массив T × N, где T – число дней, N - количество доступных акций.
- op –вектор цен открытия
- lo – вектор дневных минимумов
- cl – вектор цен закрытия

В нашем случае рассмотрим S&P 500. (Правда отметим наличие здесь ошибки выжившего (survivorship bias))

Код программы (bog.py):

```
# Example 4.1: Buy-on-Gap Model on SPX Stocks
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.formula.api as sm
import statsmodels.tsa.stattools as ts
```

```
import statsmodels.tsa.vector_ar.vecm as vm

topN=10 # Max number of positions
entryZscore=1
lookback=20 # for MA
op=pd.read_csv('inputDataOHLCDaily_20120424_op.csv')
hi=pd.read_csv('inputDataOHLCDaily_20120424_hi.csv')
lo=pd.read_csv('inputDataOHLCDaily_20120424_lo.csv')
cl=pd.read_csv('inputDataOHLCDaily_20120424_cl.csv')
stocks=pd.read_csv('inputDataOHLCDaily_20120424_stocks.csv')
op['Var1']=pd.to_datetime(op['Var1'],  format='%Y%m%d').dt.date#
remove HH:MM:SS
op.columns=np.insert(stocks.values, 0, 'Date')
op.set_index('Date', inplace=True)
hi['Var1']=pd.to_datetime(hi['Var1'],  format='%Y%m%d').dt.date#
remove HH:MM:SS
hi.columns=np.insert(stocks.values, 0, 'Date')
hi.set_index('Date', inplace=True)
lo['Var1']=pd.to_datetime(lo['Var1'],  format='%Y%m%d').dt.date#
remove HH:MM:SS
lo.columns=np.insert(stocks.values, 0, 'Date')
lo.set_index('Date', inplace=True)
cl['Var1']=pd.to_datetime(cl['Var1'],  format='%Y%m%d').dt.date#
remove HH:MM:SS
cl.columns=np.insert(stocks.values, 0, 'Date')
cl.set_index('Date', inplace=True)
stdretC2C90d=cl.pct_change().rolling(90).std().shift(1)
buyPrice=lo.shift()*(1-entryZscore*stdretC2C90d)
retGap=(op-lo.shift())/lo.shift()
ma=cl.rolling(lookback).mean()
positionsTable=np.zeros(retGap.shape)
for t in np.arange(1, cl.shape[0]):
hasData=np.where(np.isfinite(retGap.iloc[t, :]) & (op.iloc[t, :]
<buyPrice.iloc[t, :]).values & (op.iloc[t, :] >ma.iloc[t,
:]).values)
```

```
hasData=hasData[0]
if len(hasData)>0:
idxSort=np.argsort(retGap.iloc[t, hasData])
positionsTable[t, hasData[idxSort.values[np.arange(-
np.min((topN, len(idxSort))),0)]]]=1

retO2C=(cl-op)/op

pnl=np.sum(positionsTable*retO2C, axis=1) # daily P&L of the
strategy
ret=pnl/topN
(np.cumprod(1+ret)-1).plot()
plt.xlabel('May 11, 2006, to April 24, 2012')
plt.ylabel('Cumulative Returns')
plt.show()
print('APR=%f Sharpe=%f' % (np.prod(1+ret)**(252/len(ret))-1,
np.sqrt(252)*np.mean(ret)/np.std(ret)))
```

Результаты программы:

APR=0.054392 Sharpe=1.163273

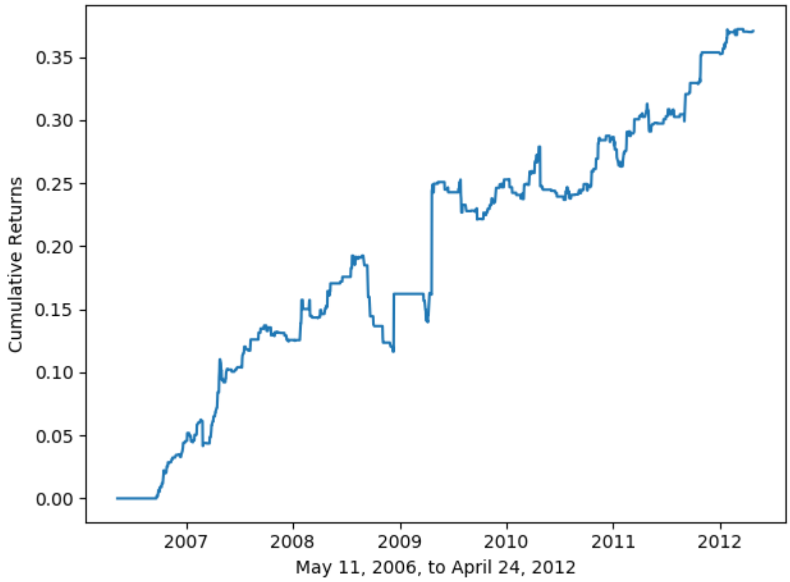


Рис 8. Кумулятивная доходность модели Buy – on – Gap на примере S&P 500.

Модель приносит 5,43% годовых в среднем с коэффициентом Шарпа 1,16

Разумеется, стоит задаться вопросом, как же мы на моменте открытия сформируем портфель, если только в этот момент появляются цены открытия, действительно никак, но зато мы можем использовать предварительные для открытия цены (например, ARCA) для определения первоначальных торговых сигналов. Сигналы, определенные таким образом, не будут точно соответствовать фактическим ценам открытия, но как правило разница пренебрежимо мала. Мы можем назвать эту разницу сигнальным шумом.

Можно модернизировать стратегию так, что она зеркально отразится: Можно совершать короткие продажи “поднявшихся вверх” акций, разрыв которых между максимумом и ценой открытия больше стандартного отклонения, но не превышает 20-ти дневной скользящей средней.

Эта стратегия довольно хорошо известна среди трейдеров, и там много вариаций на одну и ту же тему. Например, можно торговать обе версии одновременно.

Арбитраж между ETF и компонентами (акциями).

Прежде всего необходимо несколько данных: входной массив TxN, где T -количество дней, N - количество рассматриваемых акций, cl–ежедневная цена закрытия. Также, понадобится массив цен SPY (Tx1) а также значения цен закрытия cl_etfдля SPY. (Не забыть убедиться, что данные совпадают).

На основе теста Йохансена между каждой акцией в SPX и SPXпроизводим анализ на коинтеграцию на обучающем наборе, находим, что есть 98 акций, которые отдельно коинтегрированны с SPY. Теперь мы можем сформировать портфель из акций, которые по отдельности коинтегрированны с SPY с равным распределением капитала на каждую акцию (не забывая о логарифмах цен). Затем тестируем коинтеграцию этого портфеля с SPY.

Кодпрограммы (indexArb.py):

```
#Arbitrage between SPY and Its Component Stocks

import numpyas np
import pandas as pd
import matplotlib.pyplotas plt
import statsmodels.formula.apias sm
import statsmodels.tsa.stattoolsas ts
import statsmodels.tsa.vector_ar.vecmas vm
```

```
# Stocks
cl=pd.read_csv('inputDataOHLCDaily_20120424_cl.csv')
stocks=pd.read_csv('inputDataOHLCDaily_20120424_stocks.csv')

cl['Var1']=pd.to_datetime(cl['Var1'],  format='%Y%m%d').dt.date#
remove HH:MM:SS
cl.columns=np.insert(stocks.values, 0, 'Date')
cl.set_index('Date', inplace=True)

# ETFs
cl_etf=pd.read_csv('inputData_ETF_cl.csv')
etfs=pd.read_csv('inputData_ETF_stocks.csv')

cl_etf['Var1']=pd.to_datetime(cl_etf['Var1'],
format='%Y%m%d').dt.date# remove HH:MM:SS
cl_etf.columns=np.insert(etfs.values, 0, 'Date')
cl_etf.set_index('Date', inplace=True)

# Merge on common dates
df=pd.merge(cl, cl_etf, how='inner', on='Date')
cl_stocks=df[cl.columns]
cl_etf=df[cl_etf.columns]

# Use SPY only
cl_etf=cl_etf['SPY'] # This turns cl_etf into Series
trainDataIdx=df.index[(df.index>pd.datetime(2007, 1, 1).date())
& (df.index<= pd.datetime(2007, 12, 31).date())]
testDataIdx =df.index[df.index>pd.datetime(2007, 12, 31).date()]
isCoint=np.full(stocks.shape[1], False)
for s in range(stocks.shape[1]):
# Combine the two time series into a matrix y2 for input into
Johansen test
y2=pd.concat([cl_stocks.loc[trainDataIdx].iloc[:, s],
cl_etf.loc[trainDataIdx]], axis=1)
    y2=y2.loc[y2.notnull().all(axis=1),]
```

```
if (y2.shape[0] >250):
# Johansen test
result=vm.coint_johansen(y2.values, det_order=0, k_ar_diff=1)
if (result.lr1[0] >result.cvt[0,0]):
isCoint[s]=True
print(isCoint.sum())
yN=cl_stocks.loc[trainDataIdx, isCoint]
logMktVal_long=np.sum(np.log(yN), axis=1) # The net market value
of the long-only portfolio is same as the "spread"
# Confirm that the portfolio cointegrates with SPY
ytest=pd.concat([logMktVal_long,
np.log(cl_etf.loc[trainDataIdx])], axis=1)
result=vm.coint_johansen(ytest, det_order=0, k_ar_diff=1)
print(result.lr1)
print(result.cvt)
print(result.lr2)
print(result.cvm)
#Apply linear mean-reversion model on test set
yNplus=pd.concat([cl_stocks.loc[testDataIdx, isCoint],
pd.DataFrame(cl_etf.loc[testDataIdx])], axis=1)  # Array of
stock and ETF prices
weights=np.column_stack((np.full((testDataIdx.shape[0],
isCoint.sum()), result.evec[0,0]),
np.full((testDataIdx.shape[0], 1), result.evec[1, 0]))) # Array
of log market value of stocks and ETF's
logMktVal=np.sum(weights*np.log(yNplus), axis=1) # Log market
value of long-short portfolio
lookback=5
numUnits =-(logMktVal-
logMktVal.rolling(lookback).mean())/logMktVal.rolling(lookback).
std() # capital invested in portfolio in dollars.  movingAvg and
movingStd are functions from epchan.com/book2
positions=pd.DataFrame(np.expand_dims(numUnits,
axis=1)*weights)# results.evec(:, 1)' can be viewed as the
capital allocation, while positions is the dollar capital in
```

```
each ETF.
pnl=np.sum((positions.shift().values)*(np.log(yNplus)-
np.log(yNplus.shift()).values), axis=1) # daily P&L of the
strategy
ret=pd.DataFrame(pnl.values/np.sum(np.abs(positions.shift()),
axis=1).values)
(np.cumprod(1+ret)-1).plot()
plt.show()
print('APR=%f Sharpe=%f' % (np.prod(1+ret)**(252/len(ret))-1,
np.sqrt(252)*np.mean(ret)/np.std(ret)))
```

Результаты программы:

98

[15.86864835 6.19735725]

[[13.4294 15.4943 19.9349]

[2.7055 3.8415 6.6349]]

[9.6712911 6.19735725]

[[12.2971 14.2639 18.52]

[2.7055 3.8415 6.6349]]

APR=0.044930 Sharpe=1.323110

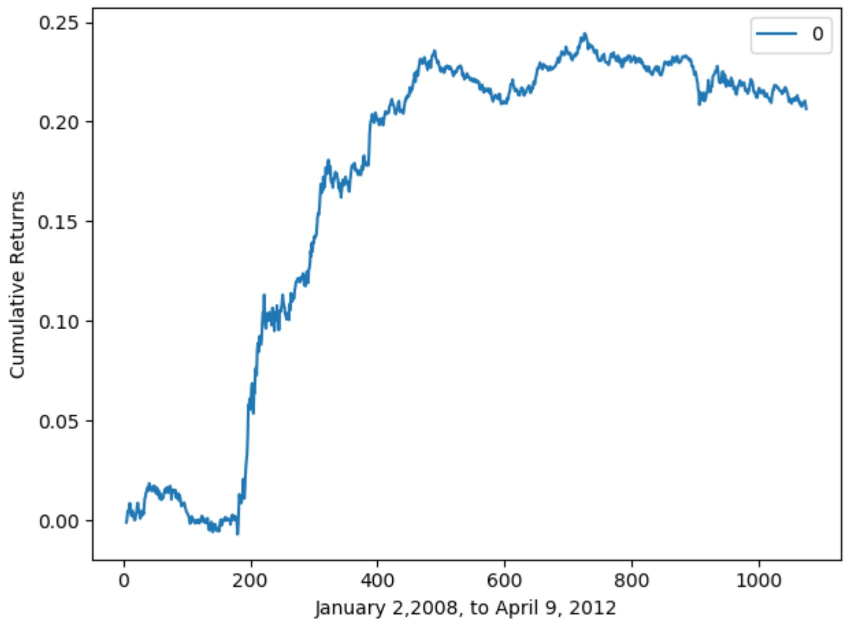


Рис 9. Кумулятивные доходы от арбитража между SPYи его компонентами.

Среднегодовой процент составил 4,5% а коэффициент Шарпа равен 1,3. Заметно, что стратегия показывала хорошие результаты ближе к началу, потому что здесь никак не учтено изменение коэффициента хеджирования акций. Для лучших показателей нужно добавлять динамическое изменение коэффициента хеджирования. Важным замечанием является тот факт, что все акции куплены, а позиция хеджируется короткой продажей ETFSPY.

Примечание:

Вообще говоря, можно задаться вопросом, почему мы просто не запустили коинтеграцию просто через тест Йохансена, протестировав все 500 акций SPX и плюс SPY, и пусть алгоритм автоматически найдёт собственный вектор коинтегрированных компонентов, включая SPY. (Не все коинтегрированные отношения из множества всех акций, составляющих ETF, и самого ETFобязательно включают SPY, но нам нужно выбрать только те варианты, где включает). Проблема с этим подходом двоякая:

1. Реализация теста Йохансена может принять максимум только 12 символов (LeSage, 1998).

2. Собственные векторы обычно включают как длинные, так и короткие позиции акций. Это означает, что у нас не может быть длинного портфеля акций, который хеджируется короткой позицией SPYили наоборот. Это проблема, потому что если у нас есть короткие позиции в портфеле акций и короткий SPYодновременно, то мы получаем двойную короткую позицию на некоторых акциях, таким образом, увеличиваются риски.

Перекрёстная средняя реверсия: Линейная Long – Short модель.

На входе получаем массив T × N и вектор cl - цены закрытия, где как обычно T - количество торговых дней, а N - количество акций в SPX.

Реализация уравнения (28) на Python

Код программы (andrewlo_2007_2012.py):

```
# Example 4.3and 4.4: Linear Long-Short Model on Stocksand
Intraday Linear Long-Short Model on Stocks

import numpy as np
import pandas as pd
import matplotlib.pyplotas plt
import statsmodels.formula.apias sm
```

```
import statsmodels.tsa.stattoolsas ts
import statsmodels.tsa.vector_ar.vecmas vm
# Stocks
cl_=pd.read_csv('inputDataOHLCDaily_20120424_cl.csv')
stocks=pd.read_csv('inputDataOHLCDaily_20120424_stocks.csv')
cl_['Var1']=pd.to_datetime(cl_['Var1'],
format='%Y%m%d').dt.date# remove HH:MM:SS
cl_.columns=np.insert(stocks.values, 0, 'Date')
cl_.set_index('Date', inplace=True)
cl_=cl_.loc[(cl_.index>= pd.datetime(2007,1, 3).date()) &
(cl_.index<= pd.datetime(2011,12,30).date()),:]

op=pd.read_csv('inputDataOHLCDaily_20120424_op.csv')
op['Var1']=pd.to_datetime(op['Var1'],  format='%Y%m%d').dt.date#
remove HH:MM:SS
op.columns=np.insert(stocks.values, 0, 'Date')
op.set_index('Date', inplace=True)
op=op.loc[(op.index>= pd.datetime(2007,1, 3).date()) &
(op.index<= pd.datetime(2011,12,30).date()),:]
ret=cl_.pct_change() # daily returns
marketRet=np.mean(ret, axis=1) # equal weighted market index
return
weights=-(np.array(ret)-np.reshape(marketRet.values,
(ret.shape[0], 1)))
weights=weights/pd.DataFrame(np.abs(weights)).sum(axis=1).values
.reshape((weights.shape[0], 1))
weights=pd.DataFrame(weights, columns=stocks.values[0],
index=np.array(ret.index))
dailyret=(weights.shift()*ret).sum(axis=1) # Capital is always
one
((1+dailyret).cumprod()-1).plot()
plt.show()
print('APR=%f Sharpe=%f' %
(np.prod(1+dailyret)**(252/len(dailyret))-1,
np.sqrt(252)*np.mean(dailyret)/np.std(dailyret)))
```

```
# APR=13.7%, Sharpe=1.3
ret=(op-cl_.shift())/cl_.shift() # daily returns
marketRet=np.mean(ret, axis=1) # equal weighted market index
return
weights=-(np.array(ret)-np.reshape(marketRet.values,
(ret.shape[0], 1)))
weights=weights/pd.DataFrame(np.abs(weights)).sum(axis=1).values
.reshape((weights.shape[0], 1))
weights=pd.DataFrame(weights, columns=stocks.values[0],
index=np.array(ret.index))
dailyret=(weights*(cl_-op)/op).sum(axis=1) # Capital is always
one
((1+dailyret).cumprod()-1).plot()
plt.show()
print('APR=%f Sharpe=%f' %
(np.prod(1+dailyret)**(252/len(dailyret))-1,
np.sqrt(252)*np.mean(dailyret)/np.std(dailyret)))
```

Результаты программы:

APR=0.136776 Sharpe=1.259979

APR=0.731553 Sharpe=4.715156

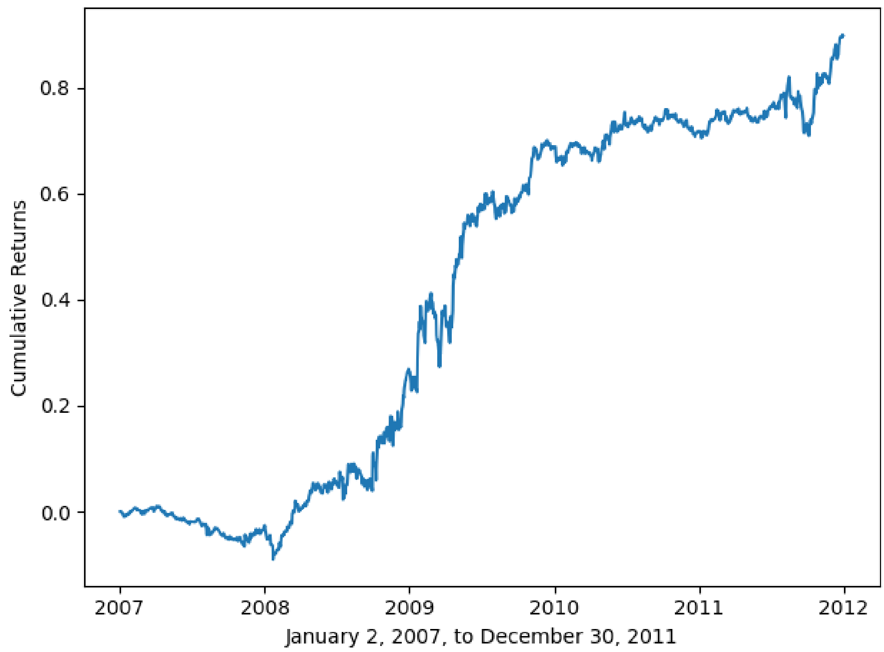


Рис 10. Накопленная доходность линейной Long – short модели.

Примечательной особенностью данной стратегии является то, что она полностью линейна, не имеет параметров и практически идеально нейтральна к ценам.

Среднегодовая доходность составила 13.7% с коэффициентом Шарпа, равным 1.3 в период с 2 января 2007 по 30 декабря 2011. (Это при условии, что бэктестирование происходит на таком индексе как SPX, обычно, бэктестирование на меньших выборках обеспечит даже большую доходность)

Можно модифицировать предыдущую стратегию для повышения доходности с помощью определения весов на момент открытия через разницу цен предыдущего закрытия и нового открытия. Все позиции ликвидируются на закрытии, тем самым превращая стратегию во внутридневную. Получаем внутридневную линейную Long-Short модель по акциям.

Соотношение среднегодовой доходности и коэффициента Шарпа за тот же период составляет 73% и 4,7, соответственно. Несмотря на такое впечатляющий результат, версия open – to – close имеет несколько недостатков, которых нет в версии close – to – close. Во-первых, транзакционные издержки (не включенные в наши бэктесты) будут удвоены, потому что мы торгуем два раза, а не только один раз в сутки. Во-вторых, так как эта стратегия также должна использовать “открытые” цены для определить торговые сигналы для входа на открытие, появляется шум торгового сигнала, который я упомянул в модели Buy – on – Gap.

На самом деле, даже для стратегии close – to – close мы также не можем использовать точную цену закрытия, поэтому берутся цены всего на несколько секунд перед закрытием, когда рынок открыт и имеет высокую ликвидность.

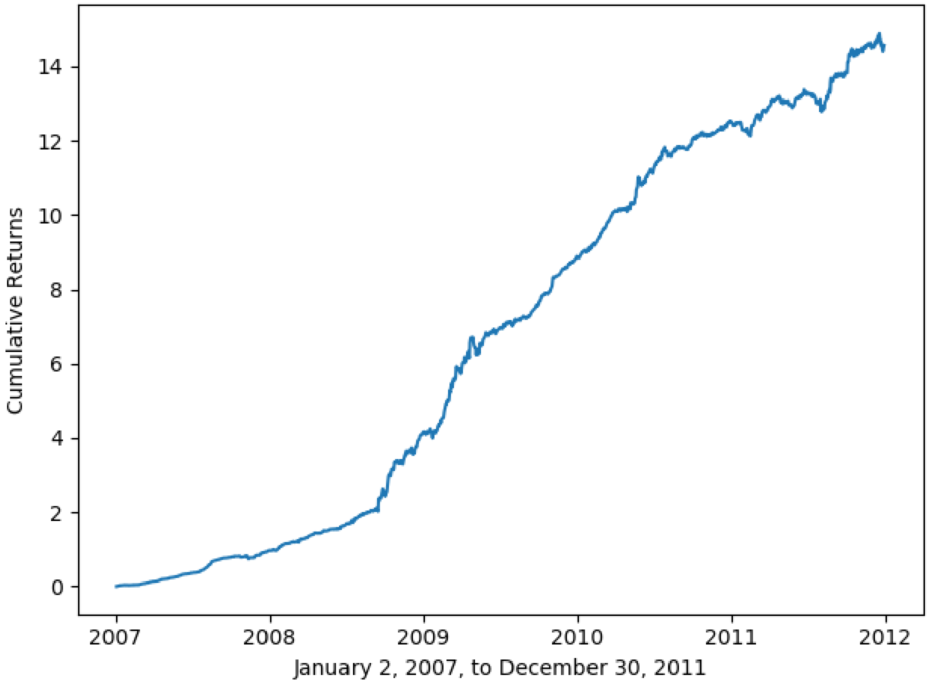


Рис 11. Накопленная доходность внутридневной линейной Long – short модели.

Примечательной особенностью данной стратегии является то, что она полностью линейна, не имеет параметров и практически идеально нейтральна к ценам.

Здесь присутствуют также дополнительные переменные, которые называются факторами, которые лучше прогнозируют в перекрёстной средней реверсии по сравнению с относительной доходностью.

Ещё одна популярная переменная, которую трейдеры используют для ранжирования акций — это соотношение цена-прибыль (P/E).

Тестирование временного ряда на импульс.

Код программы (TU_mom.py):

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.formula.api as sm
import statsmodels.tsa.stattools as ts
import statsmodels.tsa.vector_ar.vecm as vm
from scipy.stats.stats import pearsonr
```

```
df=pd.read_csv('inputDataOHLCDaily_TU_20120511.csv')
df['Date']=pd.to_datetime(df['Date'],  format='%Y%m%d').dt.date#
remove HH:MM:SS
df.set_index('Date', inplace=True)

for lookback in [1, 5, 10, 25, 60, 120, 250]:
for holddays in [1, 5, 10, 25, 60, 120, 250]:
ret_lag=df.pct_change(periods=lookback)
ret_fut=df.shift(-holddays).pct_change(periods=holddays)
if (lookback >= holddays):
indepSet=range(0, ret_lag.shape[0], holddays)
else:
indepSet=range(0, ret_lag.shape[0], lookback)

ret_lag=ret_lag.iloc[indepSet]
ret_fut=ret_fut.iloc[indepSet]
goodDates=(ret_lag.notna() &ret_fut.notna()).values
        (cc, pval)=pearsonr(ret_lag[goodDates],
ret_fut[goodDates])
print('%4i %4i %7.4f %7.4f' % (lookback, holddays, cc, pval))
lookback=250
holddays=25
longs=df >df.shift(lookback)
shorts=df <df.shift(lookback)
pos=np.zeros(df.shape)
for h in range(holddays-1):
long_lag=longs.shift(h).fillna(False)
short_lag=shorts.shift(h).fillna(False)
    pos[long_lag]=pos[long_lag]+1
pos[short_lag]=pos[short_lag]-1
pos=pd.DataFrame(pos)
pnl=np.sum((pos.shift().values)*(df.pct_change().values),
axis=1) # daily P&L of the strategy
ret=pnl/np.sum(np.abs(pos.shift()), axis=1)
```

```
cumret=(np.cumprod(1+ret)-1)
cumret.plot()
plt.show()
print('APR=%f Sharpe=%f' % (np.prod(1+ret)**(252/len(ret))-1,
np.sqrt(252)*np.mean(ret)/np.std(ret)))
from calculateMaxDDimport calculateMaxDD
maxDD, maxDDD, i=calculateMaxDD(cumret.fillna(0))
print('Max DD=%f Max DDD in days=%i' % (maxDD, maxDDD))
```

Результаты программы:

1-ый столбец – период анализа

2-ой столбец – период вложения

3-ий столбец – коэффициент корреляции

4-ый столбец – p-значение

1   1 -0.0576 0.0100
1   5 -0.0769 0.0006
1   10 -0.0274 0.2226
1   25 -0.0139 0.5369
1   60 0.0325 0.1529
1 120 0.0205 0.3735
1 250 0.0383 0.1089
5   1 -0.0755 0.0007
5   5 -0.1271 0.0111
5   10 -0.0471 0.3485
5   25 0.0323 0.5221
5   60 0.0860 0.0907
5 120 0.0543 0.2933
5 250 0.0988 0.0649
10   1 -0.0280 0.2118
10   5 -0.0485 0.3348
10   10 0.0367 0.6072
10   25 0.1160 0.1045

10	60	0.1800	0.0120
10	120	0.0967	0.1870
10	250	0.1687	0.0256
25	1	-0.0140	0.5352
25	5	0.0319	0.5272
25	10	0.1511	0.0340
25	25	0.1955	0.0843
25	60	0.2372	0.0378
25	120	0.1755	0.1320
25	250	0.2482	0.0383
60	1	0.0313	0.1685
60	5	0.0799	0.1160
60	10	0.1720	0.0165
60	25	0.1848	0.1076
60	60	0.2188	0.2213
60	120	0.0000	0.9999
60	250	0.3080	0.1040
120	1	0.0222	0.3352
120	5	0.0567	0.2731
120	10	0.0960	0.1902
120	25	0.1566	0.1796
120	60	-0.0149	0.9353
120	120	0.2179	0.4176
120	250	0.3487	0.2218
250	1	0.0411	0.0855
250	5	0.1069	0.0456
250	10	0.1788	0.0179
250	25	0.2719	0.0228
250	60	0.3282	0.0822
250	120	0.3606	0.2053

250	250	0.4896	0.2648

APR=0.016751 Sharpe=1.089553

Max DD=-0.024847 Max DDD in days=342

Мы видим, что существует некоторый компромисс между коэффициентом корреляциии p-значением. Следующие пары (в днях) представляют из себя лучшие компромиссы: (60, 10), (60, 25), (250, 10), (250, 25), (250, 60),(250, 120).

Я также проверил корреляцию между знаками прошлого и будущего.

и результаты не сильно отличаются от результатов, приведенных в таблице 6.1. Я нашел лучшей парой в этом случае являются (60, 10), (250, 10) и (250, 25) кандидатуры.

Расчёт величины максимальной просадки и самого продолжительного периода просадки.

Код программы (calculateMaxDD.py и calculateMaxDD_UnitTest.py)

```
import numpy as np
def calculateMaxDD(cumret):
highwatermark=np.zeros(cumret.shape)
    drawdown=np.zeros(cumret.shape)
drawdownduration=np.zeros(cumret.shape)
for t in np.arange(1, cumret.shape[0]):
highwatermark[t]=np.maximum(highwatermark[t-1], cumret[t])
        drawdown[t]=(1+cumret[t])/(1+highwatermark[t])-1
if drawdown[t]==0:
drawdownduration[t]=0
else:
drawdownduration[t]=drawdownduration[t-1]+1
maxDD, i=np.min(drawdown), np.argmin(drawdown) # drawdown < 0
always
maxDDD=np.max(drawdownduration)
return maxDD, maxDDD, i
```

Другой способ:

```
import numpy as np
from calculateMaxDD import calculateMaxDD
cumret=np.array([10, 9, 8, 7, 11, 9, 7, 5, 5, 12])
maxDD, maxDDD, i=calculateMaxDD(cumret)

assert(maxDD==-0.5)
assert(maxDDD==4)
assert(i==7)

print(maxDD, maxDDD, i)
```

Результаты программы:

-0.5 4.0 7

То есть maxDD = -0.5

maxDDD = 4.0

(Максимальная величина просадки составила 0.5 единиц, а самая продолжительная просадка составила 4 периода)

Пересечение скользящей средней (Реализация на примере акций Apple и окон в размере 40 и 100 дней):

```
import pandas_datareader as pdr
import numpy as np
import pandas as pd
import datetime
ticker = pdr.get_data_yahoo('AAPL',
                          start=datetime.datetime(2006, 10, 1),
                          end=datetime.datetime(2010, 1, 1))
# Initialize the short and long windows
short_window = 40
long_window = 100

# Initialize the `signals` DataFrame with the `signal` column
signals = pd.DataFrame(index=ticker.index)
signals['signal'] = 0.0

# Create short simple moving average over the short window
signals['short_mavg'] =
ticker['Close'].rolling(window=short_window, min_periods=1,
```

```
center=False).mean()

# Create long simple moving average over the long window
signals['long_mavg'] =
ticker['Close'].rolling(window=long_window, min_periods=1,
center=False).mean()

# Create signals
signals['signal'][short_window:] =
np.where(signals['short_mavg'][short_window:]
                                            >
signals['long_mavg'][short_window:], 1.0, 0.0)

# Generate trading orders
signals['positions'] = signals['signal'].diff()

# Print `signals`
print(signals)

# Import `pyplot` module as `plt`
import matplotlib.pyplot as plt

# Initialize the plot figure
fig = plt.figure()

# Add a subplot and label for y-axis
ax1 = fig.add_subplot(111, ylabel='Price in $')

# Plot the closing price
ticker['Close'].plot(ax=ax1, color='r', lw=2.)

# Plot the short and long moving averages
signals[['short_mavg', 'long_mavg']].plot(ax=ax1, lw=2.)

# Plot the buy signals
ax1.plot(signals.loc[signals.positions == 1.0].index,
         signals.short_mavg[signals.positions == 1.0],
         '^', markersize=10, color='m')

# Plot the sell signals
ax1.plot(signals.loc[signals.positions == -1.0].index,
         signals.short_mavg[signals.positions == -1.0],
         'v', markersize=10, color='k')

# Show the plot
plt.show()

print('backtesting')

# Set the initial capital
initial_capital = float(100000.0)
```

```
# Create a DataFrame `positions`
positions = pd.DataFrame(index=signals.index).fillna(0.0)

# Buy a 100 shares
positions['AAPL'] = 100 * signals['signal']

# Initialize the portfolio with value owned
portfolio = positions.multiply(ticker['Adj Close'], axis=0)

# Store the difference in shares owned
pos_diff = positions.diff()

# Add `holdings` to portfolio
portfolio['holdings'] = (positions.multiply(ticker['Adj Close'],
axis=0)).sum(axis=1)

# Add `cash` to portfolio
portfolio['cash'] = initial_capital -
(pos_diff.multiply(ticker['Adj Close'],
axis=0)).sum(axis=1).cumsum()

# Add `total` to portfolio
portfolio['total'] = portfolio['cash'] + portfolio['holdings']

# Add `returns` to portfolio
portfolio['returns'] = portfolio['total'].pct_change()

# Print the first lines of `portfolio`
print(portfolio)

plt.plot(portfolio)
```

Результат программы:

	signal	short_mavg	long_mavg	positions
Date				
2006-10-02	0.0	10.694285	10.694285	NaN
2006-10-03	0.0	10.638571	10.638571	0.0
2006-10-04	0.0	10.681905	10.681905	0.0
2006-10-05	0.0	10.683928	10.683928	0.0
2006-10-06	0.0	10.667714	10.667714	0.0
2006-10-09	0.0	10.666667	10.666667	0.0
2006-10-10	0.0	10.649184	10.649184	0.0
2006-10-11	0.0	10.625714	10.625714	0.0
2006-10-12	0.0	10.639683	10.639683	0.0
2006-10-13	0.0	10.647429	10.647429	0.0
2006-10-16	0.0	10.658701	10.658701	0.0
2006-10-17	0.0	10.654881	10.654881	0.0
2006-10-18	0.0	10.654286	10.654286	0.0
2006-10-19	0.0	10.699286	10.699286	0.0
2006-10-20	0.0	10.747429	10.747429	0.0
2006-10-23	0.0	10.803036	10.803036	0.0

```
2006-10-24     0.0   10.848655  10.848655        0.0
2006-10-25     0.0   10.894206  10.894206        0.0
2006-10-26     0.0   10.938797  10.938797        0.0
2006-10-27     0.0   10.966214  10.966214        0.0
2006-10-30     0.0   10.991088  10.991088        0.0
2006-10-31     0.0   11.017987  11.017987        0.0
2006-11-01     0.0   11.030621  11.030621        0.0
2006-11-02     0.0   11.041131  11.041131        0.0
2006-11-03     0.0   11.046857  11.046857        0.0
2006-11-06     0.0   11.059945  11.059945        0.0
2006-11-07     0.0   11.076296  11.076296        0.0
2006-11-08     0.0   11.101378  11.101378        0.0
2006-11-09     0.0   11.129113  11.129113        0.0
2006-11-10     0.0   11.153952  11.153952        0.0
          ...     ...       ...       ...
2009-11-18     1.0   27.701143  24.980129        0.0
2009-11-19     1.0   27.760750  25.063100        0.0
2009-11-20     1.0   27.823429  25.144657        0.0
2009-11-23     1.0   27.893893  25.238743        0.0
2009-11-24     1.0   27.961964  25.332786        0.0
2009-11-25     1.0   28.029250  25.431057        0.0
2009-11-27     1.0   28.099714  25.521586        0.0
2009-11-30     1.0   28.153321  25.612371        0.0
2009-12-01     1.0   28.192429  25.695871        0.0
2009-12-02     1.0   28.214643  25.772857        0.0
2009-12-03     1.0   28.236893  25.850300        0.0
2009-12-04     1.0   28.251357  25.916643        0.0
2009-12-07     1.0   28.245929  25.975829        0.0
2009-12-08     1.0   28.242571  26.030286        0.0
2009-12-09     1.0   28.270357  26.094414        0.0
2009-12-10     1.0   28.288714  26.158586        0.0
2009-12-11     1.0   28.303393  26.212771        0.0
2009-12-14     1.0   28.335286  26.268714        0.0
2009-12-15     1.0   28.350679  26.317543        0.0
2009-12-16     1.0   28.337357  26.367443        0.0
2009-12-17     1.0   28.290714  26.412957        0.0
2009-12-18     1.0   28.255821  26.463529        0.0
2009-12-21     1.0   28.235429  26.514157        0.0
2009-12-22     1.0   28.227857  26.566971        0.0
2009-12-23     1.0   28.244750  26.617929        0.0
2009-12-24     1.0   28.304179  26.680057        0.0
2009-12-28     1.0   28.358679  26.746486        0.0
2009-12-29     1.0   28.432250  26.811043        0.0
2009-12-30     1.0   28.512000  26.876943        0.0
2009-12-31     1.0   28.590500  26.942671        0.0
[819 rows x 4 columns]
backtesting
                AAPL     holdings    ...            total   returns
Date                                 ...
2006-10-02     0.000000     0.000000    ...     100000.000000       NaN
2006-10-03     0.000000     0.000000    ...     100000.000000  0.000000
```

2006-10-04	0.000000	0.000000	...	100000.000000	0.000000
2006-10-05	0.000000	0.000000	...	100000.000000	0.000000
2006-10-06	0.000000	0.000000	...	100000.000000	0.000000
2006-10-09	0.000000	0.000000	...	100000.000000	0.000000
2006-10-10	0.000000	0.000000	...	100000.000000	0.000000
2006-10-11	0.000000	0.000000	...	100000.000000	0.000000
2006-10-12	0.000000	0.000000	...	100000.000000	0.000000
2006-10-13	0.000000	0.000000	...	100000.000000	0.000000
2006-10-16	0.000000	0.000000	...	100000.000000	0.000000
2006-10-17	0.000000	0.000000	...	100000.000000	0.000000
2006-10-18	0.000000	0.000000	...	100000.000000	0.000000
2006-10-19	0.000000	0.000000	...	100000.000000	0.000000
2006-10-20	0.000000	0.000000	...	100000.000000	0.000000
2006-10-23	0.000000	0.000000	...	100000.000000	0.000000
2006-10-24	0.000000	0.000000	...	100000.000000	0.000000
2006-10-25	0.000000	0.000000	...	100000.000000	0.000000
2006-10-26	0.000000	0.000000	...	100000.000000	0.000000
2006-10-27	0.000000	0.000000	...	100000.000000	0.000000
2006-10-30	0.000000	0.000000	...	100000.000000	0.000000
2006-10-31	0.000000	0.000000	...	100000.000000	0.000000
2006-11-01	0.000000	0.000000	...	100000.000000	0.000000
2006-11-02	0.000000	0.000000	...	100000.000000	0.000000
2006-11-03	0.000000	0.000000	...	100000.000000	0.000000
2006-11-06	0.000000	0.000000	...	100000.000000	0.000000
2006-11-07	0.000000	0.000000	...	100000.000000	0.000000
2006-11-08	0.000000	0.000000	...	100000.000000	0.000000
2006-11-09	0.000000	0.000000	...	100000.000000	0.000000
2006-11-10	0.000000	0.000000	...	100000.000000	0.000000
...	...	...		...	...
2009-11-18	1961.924934	1961.924934	...	101156.712532	-0.000098
2009-11-19	1910.009384	1910.009384	...	101104.796982	-0.000513
2009-11-20	1904.389000	1904.389000	...	101099.176598	-0.000056
2009-11-23	1961.162949	1961.162949	...	101155.950546	0.000562
2009-11-24	1947.445869	1947.445869	...	101142.233467	-0.000136
2009-11-25	1945.064545	1945.064545	...	101139.852142	-0.000024
2009-11-27	1910.771370	1910.771370	...	101105.558968	-0.000339
2009-11-30	1904.293442	1904.293442	...	101099.081039	-0.000064
2009-12-01	1876.288605	1876.288605	...	101071.076202	-0.000277
2009-12-02	1869.239235	1869.239235	...	101064.026833	-0.000070
2009-12-03	1871.620941	1871.620941	...	101066.408539	0.000024
2009-12-04	1841.519547	1841.519547	...	101036.307144	-0.000298
2009-12-07	1799.891663	1799.891663	...	100994.679260	-0.000412
2009-12-08	1808.655357	1808.655357	...	101003.442955	0.000087
2009-12-09	1884.194756	1884.194756	...	101078.982353	0.000748
2009-12-10	1871.144676	1871.144676	...	101065.932274	-0.000129
2009-12-11	1854.378510	1854.378510	...	101049.166107	-0.000166
2009-12-14	1876.383209	1876.383209	...	101071.170807	0.000218
2009-12-15	1849.616814	1849.616814	...	101044.404411	-0.000265
2009-12-16	1857.808495	1857.808495	...	101052.596092	0.000081
2009-12-17	1827.611732	1827.611732	...	101022.399330	-0.000299
2009-12-18	1861.618614	1861.618614	...	101056.406212	0.000337

2009-12-21	1888.290977	1888.290977	...	101083.078575	0.000264
2009-12-22	1908.580780	1908.580780	...	101103.368378	0.000201
2009-12-23	1925.155640	1925.155640	...	101119.943237	0.000164
2009-12-24	1991.264343	1991.264343	...	101186.051941	0.000654
2009-12-28	2015.745735	2015.745735	...	101210.533333	0.000242
2009-12-29	1991.835976	1991.835976	...	101186.623573	-0.000236
2009-12-30	2016.031075	2016.031075	...	101210.818672	0.000239
2009-12-31	2007.363129	2007.363129	...	101202.150726	-0.000086

[819 rows x 5 columns]

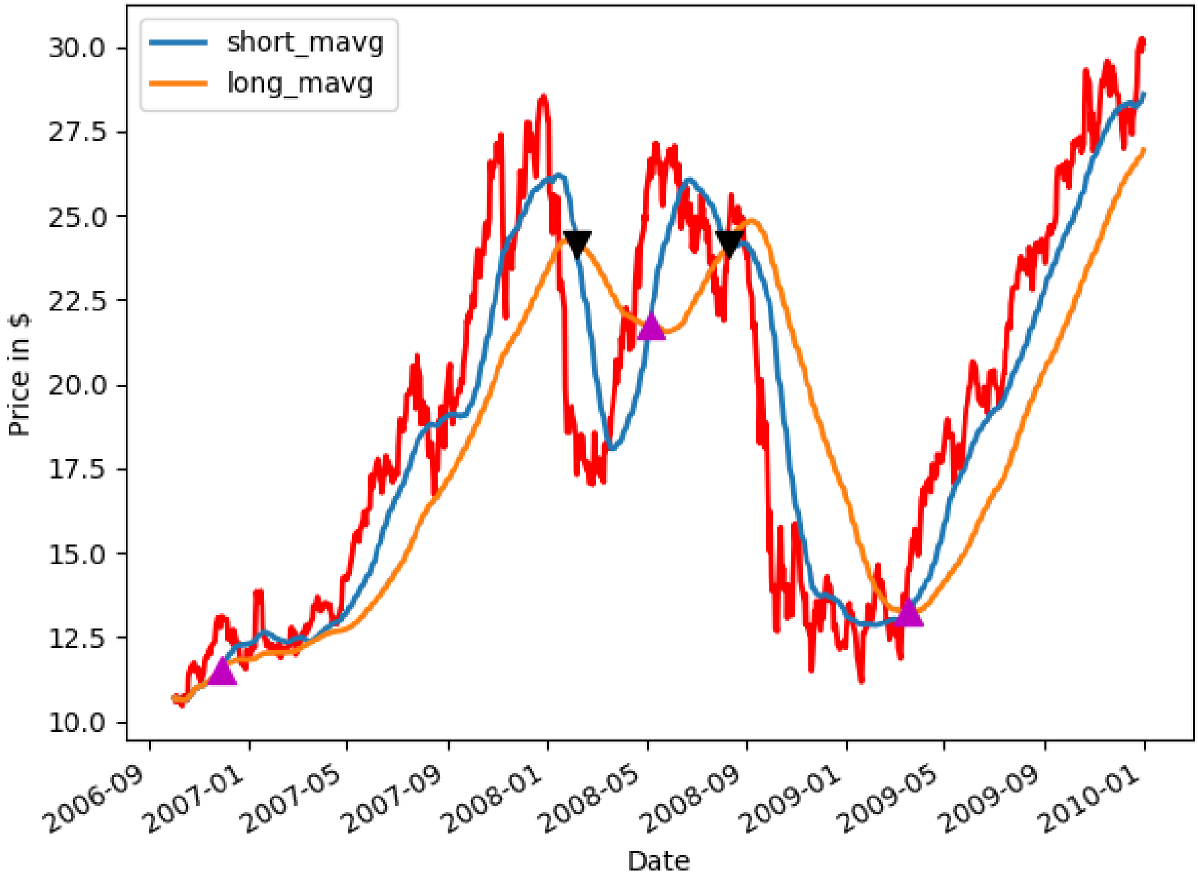


Рис 12. Сигналы, полученные с помощью пересечения скользящей средней. Синий график соответствует скользящему среднему за 40 дней, а оранжевый график соответствует скользящему среднему за 100 дней.

Итоги.

Главной моей задачей было рассмотрение различных алгоритмов, стратегий, а также их реализации. Были продемонстрированы различные подходы к тому, как можно выстраивать стратегию. В большей степени весь этот комплекс представляет теоретический и гораздо более практический интерес. Как было анонсировано в самом начале, в плане – происходит поиск временного ряда, на нём проверяются полезные свойства, такие как стационарность, что в свою очередь рождает коинтеграцию. Если ряд обладает такими свойствами, то группа стратегий возвращения к среднему – хороший вариант для дальнейшего выстраивания своей стратегии, если не выполняются свойства, то возможно два варианта, либо отрицание стационарности может означать трендовость, в таком случае группа импульсных стратегий является хорошим вариантом для дальнейшего выстраивания стратегии, также возможна ситуация и отсутствия трендовости и стационарности, тогда стоит обращать внимания на стратегии, не связанные с реверсией среднего или импульсом. Далее стоит задуматься об управлении рисками и способами, как это можно сделать, после чего внедрить эти методы в конкретную стратегию. После того, как стратегия проработана в теории, можно проверить её на практике, то есть совершить её бэктестинг на реальных данных. Для анализа бэктестированных данных следует подсчитывать как минимум два показателя – это среднегодовая доходность и коэффициент Шарпа, который является хорошей оценкой уровня риска. После того, как весь процесс пройден, можно заключить, рентабельна стратегия или нет, и, соответственно, стоит ли в неё вкладываться.

Библиографический список:

1. Hendershott, T., Jones, C. M., & Menkveld, A. J. (2011). Does algorithmic trading improve liquidity?. *The Journal of Finance*, *66*(1), 1-33. https://doi.org/10.1111/j.1540-6261.2010.01624.x

2. Hendershott, T., & Riordan, R. (2013). Algorithmic trading and the market for liquidity. *Journal of Financial and Quantitative Analysis*, *48*(4), 1001-1024. https://doi.org/10.1017/S0022109013000471

3. Johansen, S. (1988). Statistical analysis of cointegration vectors. *Journal of economic dynamics and control*, *12*(2-3), 231-254. https://doi.org/10.1016/0165-1889(88)90041-3

4. Chaboud, A. P., Chiquoine, B., Hjalmarsson, E., & Vega, C. (2014). Rise of the machines: Algorithmic trading in the foreign exchange market. *The Journal of Finance*, *69*(5), 2045-2084. https://doi.org/10.1111/jofi.12186

5. Stambaugh, R. F., Yu, J., & Yuan, Y. (2012). The short of it: Investor sentiment and anomalies. *Journal of Financial Economics*, *104*(2), 288-302. https://doi.org/10.1016/j.jfineco.2011.12.001

6. Moskowitz, T. J., Ooi, Y. H., & Pedersen, L. H. (2012). Time series momentum. *Journal of financial economics*, *104*(2), 228-250. https://doi.org/10.1016/j.jfineco.2011.11.003

7. Balvers, R. J., & Wu, Y. (2006). Momentum and mean reversion across national equity markets. *Journal of Empirical Finance*, *13*(1), 24-48. https://doi.org/10.1016/j.jempfin.2005.05.001

8. Ledoit, O., & Wolf, M. (2008). Robust performance hypothesis testing with the Sharpe ratio. *Journal of Empirical Finance*, *15*(5), 850-859. https://doi.org/10.1016/j.jempfin.2008.03.002

9. Avellaneda, M., & Lee, J. H. (2010). Statistical arbitrage in the US equities market. *Quantitative Finance*, *10*(7), 761-782. https://doi.org/10.1080/14697680903124632

10. Chan, E. (2013). *Algorithmic trading: winning strategies and their rationale* (Vol. 625). John Wiley & Sons. https://doi.org/10.1002/9781118676998

11. Poterba, J. M., & Summers, L. H. (1988). Mean reversion in stock prices: Evidence and implications. *Journal of financial economics*, *22*(1), 27-59. https://doi.org/10.1016/0304-405X(88)90021-9

12. Lakner, P. (1998). Optimal trading strategy for an investor: the case of partial information. *Stochastic Processes and their Applications*, *76*(1), 77-97. https://doi.org/10.1016/S0304-4149(98)00032-5

13. Gârleanu, N., & Pedersen, L. H. (2013). Dynamic trading with predictable returns and transaction costs. *The Journal of Finance*, *68*(6), 2309-2340. https://doi.org/10.1111/jofi.12080

14. Brogaard, J., Hendershott, T., & Riordan, R. (2014). High-frequency trading and price discovery. *The Review of Financial Studies*, *27*(8), 2267-2306. https://doi.org/10.1093/rfs/hhu032

15. Carrion, A. (2013). Very fast money: High-frequency trading on the NASDAQ. *Journal of Financial Markets*, *16*(4), 680-711. https://doi.org/10.1016/j.finmar.2013.06.005

16. Chan, E. (2009). *Quantitative trading: how to build your own algorithmic trading business* (Vol. 430). John Wiley & Sons. https://doi.org/10.1002/9781119203377

17. Taylor, A. M. (2001). Potential pitfalls for the purchasing-power-parity puzzle? Sampling and specification biases in mean-reversion tests of the law of one price. *Econometrica*, *69*(2), 473-498. https://doi.org/10.1111/1468-0262.00199

18. Chaudhuri, K., & Wu, Y. (2003). Random walk versus breaking trend in stock prices: Evidence from emerging markets. *Journal of Banking & Finance*, *27*(4), 575-592. https://doi.org/10.1016/S0378-4266(01)00252-7

19. Balvers, R., Wu, Y., & Gilliland, E. (2000). Mean reversion across national stock markets and parametric contrarian investment strategies. The Journal of Finance, 55(2), 745-772. https://doi.org/10.1111/0022-1082.00225

20. Thorp, E. O. (2011). The Kelly criterion in blackjack sports betting, and the stock market. In *The Kelly Capital Growth Investment Criterion: Theory and Practice* (pp. 789-832). https://doi.org/10.1016/B978-044453248-0.50015-0

Приложение 1.

Подробная структура работы