А.О. Глёкин

Распознавание паттернов в искусственных рыночных данных

Аннотация. В работе предлагается метод построения системы прогнозирования финансовых временных рядов в ситуации поступления новой информации на рынок. Система строится для искусственных рыночных данных, полученных путем моделирования реакции рынка на поступившую новость. Задача прогнозирования формулируется в виде задачи классификации. Для классификации используются методы машинного обучения такие как, метод бэггинга, генетический алгоритм и метод сравнения с эталонными векторами в метрике Хемминга. Применение предложенного в данной работе алгоритма на реальных финансовых временных рядах показало его состоятельность.

Ключевые слова: многоагентные модели, искусственный финансовый рынок, классификация, бэггинг, генетический алгоритм.

Введение

В работе предлагается исследовать явление поступления новостей на финансовые рынки. На Рис.1 представлена картина поведения цен и объемов торгов на нефтяной фьючерс Light Sweet Crude Oil на бирже CME в ответ на новость по запасам нефти в США, публикуемой, как правило, в 10:30 (UTC-05:00) в среду [11]. В отсутствии новости цена изменяется в диффузионном режиме и объем торгов относительно низкий. При поступлении новости цена меняется скачком и затем медленно переходит в начальный режим поведения. Процесс резкого изменения цены сопровождается значительным увеличением объемов торгов. Подобное поведение типично для всех финансовых рынков и хорошо известно участникам торгов.

Здесь рассматривается задача построения механизма анализа данных генерируемых моделью, предложенной в статье [10]. Данная модель построена на основе нелинейного марковского процесса и симулирует поведение цены и объема торгов в момент поступления новой информации на рынок. В случайные моменты времени генерируются резкие всплески цены и объемов. На Рис. 2 представлена картина поведения цен и объемов в изучаемой модели.

Задача поиска момента окончания быстрого движения цены и начало возвращения ее к исходному диффузионному поведению (момент разворота цены) чрезвычайно актуальна для участников торгов.

В данной работе предлагается метод поиска таких точек на графике цены с использованием методов машинного обучения.

Из-за природы анализируемых данных задача построения классификатора является слабо обусловленной. Предполагается наличие шума и высокой размерности пространства признаков классифицируемых объектов. В таких условиях найти классификатор с хорошей обобщающей способностью становится весьма проблематично.

Для классификации бинарных векторов здесь применяется метод минимального расстояния до эталонных векторов в метрике Хемминга. В качестве алгоритма поиска эталонных векторов, по ряду причин, удобно использовать генетический алгоритм [9]:

– обучающее множество имеет значительный шум;

– большое пространство признаков;

– эталонные векторы могут быть легко представлены в форме хромосом.

Метод бэггинга [1], предложенный Л. Брейменом (Breiman, L.) в 1996г., оказался

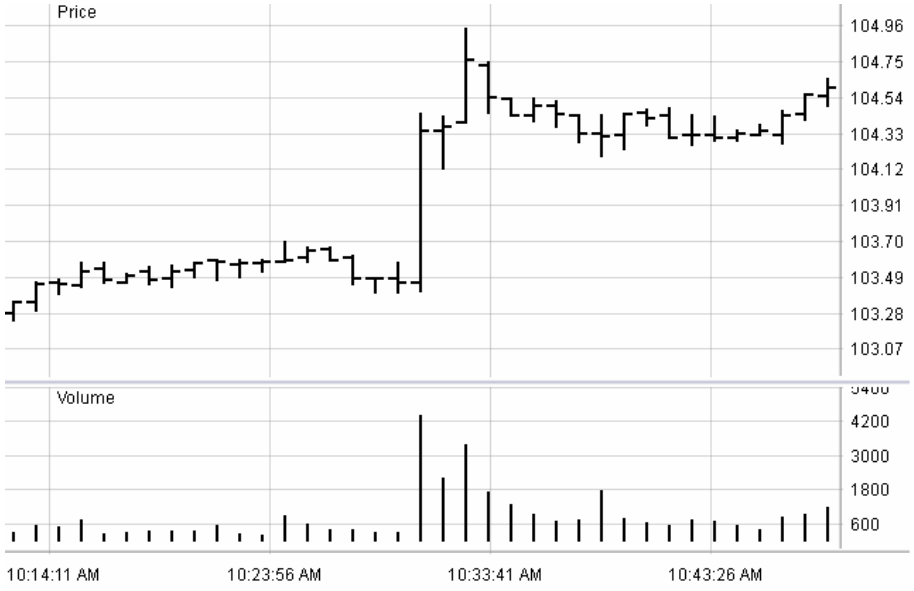

Рис.1. Цены и объемы торгов на фьючерс Light Sweet Crude Oil (CLK8) за 26.03.2008 (UTC-5:00). Минутные бары

Рис. 2. Пример поведения цен и объемов торгов в модельных данных

эффективен на малых обучающих выборках и использует нестабильные индивидуальные классификаторы в качестве компонентов комбинированного классификатора. Нестабильность классификатора означает, что при малом изменении в обучающей выборке происходит сильное изменение в индивидуальном классификаторе. В условиях большого числа признаков, малого размера выборки и высокой степени зашумленности данных, генетический алгоритм обучения ведет себя крайне неустойчиво.

1. Формализация задачи

Имеется некоторое количество исторических данных для цен и объемов сделок за фиксированный интервал времени (бары цен и объемов). Бар – одно из классических графических представлений ценовой динамики финансовых инструментов в биржевой индустрии. Все бары имеют фиксированный временной период Δ и представляют собой набор из пяти атрибутов

$$b_t = \{H_t, L_t, O_t, C_t, V_t\},$$

где t – время закрытия бара,

H_t – максимальная цена совершенной сделки,

L_t – минимальная цена совершенной сделки,

O_t – цена открытия или цена первой сделки внутри бара,

C_t – цена закрытия, или цена последней совершенной сделки внутри бара,

V_t – суммарный объем торгов.

Бары упорядочены по времени их появления.

Каждому бару b_τ ставится в соответствие число $y_\tau = C_\tau - C_{\tau+t_0}$, т.е. разность цен закрытия текущего бара b_τ и бара отстоящего на время t_0 (порядка нескольких баров) от него вперед $b_{\tau+t_0}$. Величина y_τ имеет смысл предполагаемого дохода, полученного при продаже финансового инструмента по цене C_τ, и обратном его выкупе по цене $C_{\tau+t_0}$.

Шаблоны для классификации строятся на основе массива баров $B_\tau = \{b_t : t = \tau, \tau-\Delta, \ldots \tau-(w-1)\Delta\}$, где w – размер окна (количество баров), τ – момент времени, на который строится шаблон. Далее примем $\Delta = 1$.

Задача поиска момента разворота цены рассматривается как задача классификации баров на два класса: “1” – момент разворота движения цены после роста в ответ на новость, “0” – иное поведение цены. Моментом разворота будем считать те бары, для которых выполняется условие $y_\tau - c > 0$, где константа c имеет смысл транзакционных затрат при операциях купли-продажи. Поскольку здесь рассматриваются моменты разворота движения цены после резкого роста, то бары, для которых цена H_τ не является максимальной за предыдущие $w-1$ баров, автоматически будут классифицироваться как “0”.

Шаблон цены формируется путем наложения картины цен баров на решетку P_{ij} размером $w \times r_p$, где $i = 0,1,\ldots w-1$, $j = 0,1,\ldots r_p - 1$, а P_{ij} принимает значения 0 или 1. Пример отображения картины цен баров на решетку P_{ij} представлен на Рис. 3. Закрашенные клетки соответствуют 1, не закрашенные 0 в решетке P_{ij}.

Обозначим за $[]^-$ – функцию округления числа к меньшему целому, а через $[]^+$ – функцию округления числа к большему целому числу, Δ_p – шаг дискретизации цены. Шаблон P_{ij} цены строится по следующей схеме.

1. Вычислим для каждого бара из B_τ диапазон закрашиваемых клеток в P_{ij}:

$$l_t = [(H_\tau - H_t)/\Delta_p]^-,\ d_t = [(H_t - L_t)/\Delta_p]^+,$$

где $t = \tau, \tau-1, \ldots, \tau-(w-1)$,

2. Тогда, $P_{ij} = \begin{cases} 1, \text{если } l_{\tau-i} \le j \le d_{\tau-i}; \\ 0, \text{иначе,} \end{cases}$

где $i = 0,1,\ldots w-1$, $j = 0,1,\ldots r_p - 1$.

Размер решетки $w \times r_p$ и шаг дискретизации Δ_p выбираются так, что бы уместить в себе характерное движение цены от момента поступления новости до момента разворота.

Шаблон объемов формируется путем наложения картины объемов баров на решетку V_{ij} размером $w \times r_v$, где $i = 0,1,\ldots w-1$, $j = 0,1,\ldots r_v - 1$, V_{ij} принимает значения 0 или 1. Пример отображения картины объемов баров на решетку V_{ij} представлен на Рис. 4. Закрашенные клетки соответствуют 1, не закрашенные 0 в решетке V_{ij}.

Обозначим за Δ_v – шаг дискретизации объема. Шаблон объемов V_{ij} строится по следующей схеме:

1. Вычислим для каждого бара из B_τ диапазон закрашиваемых клеток в V_{ij}:

$$h_t = [V_t / \Delta_v]^+, \text{ где } t = \tau, \tau-1, \ldots, \tau-(w-1).$$

2. Тогда, $V_{ij} = \begin{cases} 1, \text{если } j \le h_{\tau-i}; \\ 0, \text{иначе,} \end{cases}$

где $i = 0,1,\ldots w-1$, $j = 0,1,\ldots r_v - 1$.

Размер решетки $w \times r_v$ и шаг дискретизации Δ_v выбираются так, что бы уместить в себе характерные объемы торгов от момента поступления новости до момента разворота.

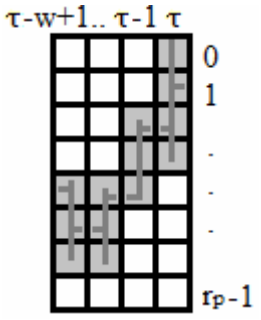

Рис. 3. Пример отображения цен баров на решетку

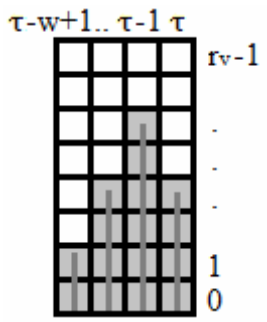

Рис. 4. Пример отображения объемов баров на решетку

Таким образом, мы отобразили рыночные данные на решетки P_{ij} и V_{ij}. Далее, полученные решетки преобразуем в двоичные векторы:

$pr_{i*r_p+j} = P_{ij}$, $i = 0,1,...w-1$, $j = 0,1,...r_p - 1$;

$vl_{i*r_v+j} = V_{ij}$, $i = 0,1,...w-1$, $j = 0,1,...r_v - 1$.

В итоге, массив баров $B_\tau = \{b_t : t = \tau, \tau - \Delta, ..., \tau - (w-1)\Delta\}$ представляется в виде пары векторов $(pr, pl)_\tau$, которой соответствует бар b_τ и, следовательно, число y_τ.

2. Задача классификации. Обучение классификатора

В этом разделе вначале ставится задача классификации бинарных векторов, а затем рассматривается задача обучения классификатора.

2.1. Вероятностная постановка задачи классификации.

Пусть [6] известна совместная плотность распределения вероятностей $\rho(x,y) = \rho_x(x)\rho(y|x)$, на парах (x,y) где $x \in X$ и число $y \in R$ есть отклик на признак x. Множество X будем называть пространством признаков, и в нашем случае оно представляет бинарные вектора длины dim, $X = \{0,1\}^{\dim}$. Задача классификации заключается в построении классификатора $d : X \to \{0,1\}$ из некоторого класса D и удовлетворяющее некоторому условию оптимальности. Класс D, в нашем случае, это множество всех правил классификации, возникающих из метода минимального расстояния до эталона класса в метрике Хемминга.

Пусть D^* есть множество всевозможных правил классификации пространства X на два класса и $D \in D^*$. В качестве критерия оптимальности правила d используется следующая функция:

$$f(d,A) = \sum_{x \in A} \rho_x(x) d(x) E_\rho(y - c \mid x) \qquad (1)$$

где c неотрицательная константа, $E_\rho(y - c \mid x)$ – условное математическое ожидание, т.е. математическое ожидание величины $y - c$ при заданном x. Подмножество $A \subseteq X$ такое, что $A = \text{supp}\, \rho_x(\bullet)$. Решением такой задачи на всем множестве классификаторов D^* является

$$d^* = arg \max_{d \in D^*} f(d,A) \qquad (2)$$

Очевидно, что

$$d^*(x) = \begin{cases} 1, \text{если } E_\rho(y - c \mid x) > 0; \\ 0, \text{иначе,} \end{cases} \qquad (3)$$

для $\forall x \in X$.

Правило d^* означает, что если для некоторого x математическое ожидание отклика y больше константы c, то такой x принадлежит классу “1”, в ином случае x принадлежит классу “0”. Аналогично определяется оптимальное решающее правило из класса D

$$d^0 = arg \max_{d \in D} f(d,A). \qquad (4)$$

В качестве меры близости некоторого классификатора d к оптимальному классификатору d^* используется разность

$$R(d,A) = f(d^*,A) - f(d,A) = \\ = \sum_{x \in A} \{\rho_x(x)(d^*(x) - d(x)) E_\rho(y - c \mid x)\}. \qquad (5)$$

Функция $R(d,A)$ имеет смысл ошибки классификации правила d. Искомый классификатор d^0 является ближайшим к оптимальному классификатору d^*

$$d^0 = \arg\min_{d\in D} R(d,A). \quad (6)$$

2.2. Обучение по выборке

На практике отсутствует информация о вероятностной мере $\rho(x,y)$, что означает невозможность вычислить значение функций $f(d,A)$ и $R(d,A)$. Задача обучения классификатора заключается в построении правила классификации близкого к оптимальному правилу d^* на основе имеющейся обучающей выборке $S=\{(x_i,y_i)\}_{i=1}^N$, N равно числу обучающих примеров.

Из (1) получаем эмпирическую функцию эффективности классификатора

$$\overline{f}(d,S)=\frac{1}{N}\sum_{i=1}^{N} d(x_i)(y_i-c). \quad (7)$$

Функция $\overline{f}(d,S)$ представляет собой оценку эффективности правила d на выборке S. Тогда, обучение классификатора на выборке S есть, по сути, поиск правила с максимальным значением функции $\overline{f}(d,S)$.

3. Бэггинг. Агрегированный классификатор

Бэггинг – метод создания агрегированного классификатора, позволяющий значительно повысить обобщающую способность классификации. Суть метода бэггинга заключается в создании ряда индивидуальных классификаторов, обученных на различных пересекающихся подвыборках исходной обучающей выборки, с последующей агрегацией полученных классификаторов в один классификатор, в котором решение принимается путем голосования. Для создания выборок, на которых обучаются индивидуальные классификаторы, используется метод бутстрапа (bootstrap) [3]. Пусть S исходная выборка размера N, тогда подвыборка S' получается путем равновероятного случайного выбора, с возвращением, N обучающих примеров из исходной выборки S. Примем для операции бутстрапа следующую запись $S' = bootstrap(S)$.

Для фиксированной выборки S, введем функцию

$$\delta(x)=Prob\{d(x)=1\},\quad \text{где } d=G(bootstrap(S)), \quad (8)$$

и означающую вероятность того, что правило d, полученное после обучения алгоритмом G на выборке S с применением метода бутстрапа, вернет 1 в ответ на признак $x\in X$.

Рассмотрим следующее правило:

$$d_\theta(x)=\begin{cases}1, \text{если } \delta(x)>\theta;\\ 0, \text{если } \delta(x)\le\theta.\end{cases} \quad (9)$$

Обозначим через $U\subset X$ множество всех таких точек x, для которых $d_\theta = d^*(x)$. Соответственно, для дополнения множества U выполняется $d_\theta \neq d^*(x)$, для $\forall x\in \overline{U}$.

Одним из методов описания принципа работы бэггинга является декомпозиция ошибки классификации на сумму смещения и отклонения классификации (bias-variance decomposition). По аналогии с определениями приведенными в работе Бреймана [2], дадим определения смещения и отклонения правила классификации для задачи рассматриваемой в этой статье.

Определение 1. Смещением правила d называется $Bias(d)=R(d,U)$. (10)

Определение 2. Отклонением правила d называется $Var(d)=R(d,\overline{U})$. (11)

Тогда меру R можно представить в виде суммы

$$R(d,X)=Bias(d)+Var(d). \quad (12)$$

Следующие свойства очевидны:

1. $Bias(d^*)=0$.
2. $Var(d^*)=0$.
3. $Var(d_\theta)=0$.

Следовательно,

$$R(d_\theta,X)=Bias(d_\theta) \quad (13)$$

Получаем, что для правила d_θ значение функции R равно смещению. Неустойчивость алгоритма обучения характеризуется большим значением отклонения Var и малым смещением

$Bias$ получаемого индивидуального классификатора. При этом смещение правила d_θ может быть выше, чем смещение индивидуального классификатора полученного после обучения на всей выборке S. Уменьшить смещение классификатора d_θ можно путем настройки параметра порога θ. Оптимальный порог θ^* для правила d_θ определяется по формуле:

$$\theta^* = \arg\min_{\theta\in[0,1]} Bias(d_\theta). \qquad (14)$$

Метод бэггинга, исходя из разложения функции R на сумму смещения и отклонения, заключается в создании агрегированного правила d_{agr} аппроксимирующего правило d_θ с целью значительно уменьшить отклонение $Var(d_{agr})$. Агрегированное правило d_{agr} строится следующим образом.

1. Вначале создается набор индивидуальных классификаторов $d_i = G(bootstrap(S))$, где $i = 1,...,B$.

2. После создания набора классификаторов $\{d_i\}_1^B$, выполняется их агрегация

$$\text{Margin}(x) = \frac{1}{B}\sum_{i=1}^{B} d_i(x), \qquad (15)$$

$$d_{agr}(x) = \begin{cases} 1, \text{если Margin}(x) > \theta, \\ 0, \text{иначе}, \end{cases} \qquad (16)$$

где параметр порога θ может принимать значения в интервале [0,1]. Значение $\text{Margin}(x)$ есть эмпирическая оценка вероятности $\delta(x)$.

При вычислении эффективности $\overline{f}(d_{agr}, S)$ важно учитывать ее статистическую достоверность. Достоверность оценки зависит от количества слагаемых входящих в сумму $\overline{f}(d,S) = \frac{1}{N}\sum_{i=1}^{N} d(x_i)(y_i - c)$, для которых $d(x_i) = 1$. Ясно, что чем меньше значение порога θ, тем для большего числа точек агрегированный классификатор принимает значение 1. Это замечание означает, что порог θ рекомендуется выбирать левее максимума функции $\overline{f}(d_{agr}, S)$. Для этого, поведение функции $\overline{f}(d_{agr}, S)$ в районе максимума должно иметь слабую зависимость от значения порога θ. Этот факт продемонстрирован на Рис. 5. Максимум функции эффективности $\overline{f}(d_{agr}, T)$ на тестовой выборке T в зависимости от порога θ лежит левее, чем максимум функции эффективности $\overline{f}(d_{agr}, S)$ на обучающей выборке S.

4. Элементарный классификатор

Элементарный классификатор содержит две группы эталонных векторов $E_0 = \{e_1, e_2, ..., e_n\}$ и $E_1 = \{e_{n+1}, e_{n+2}, ..., e_{n+m}\}$, и представляет собой набор из $n+m$ двоичных векторов размерности dim, $E = \{e_1, e_2, ..., e_{n+m}\}$. Обозначим за $h(\cdot,\cdot)$ расстояние Хемминга, тогда решающее правило элементарного классификатора будет иметь следующий вид:

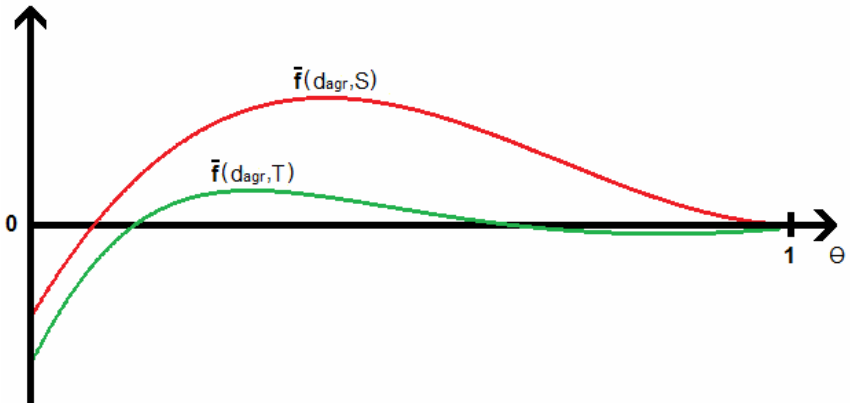

Рис. 5. Характерная зависимость функции эффективности от параметра θ на тестовой (T) и обучающей (S) выборках

$$d(x)=\begin{cases}1, \text{если } \min\limits_{e\in E_1} h(x,e) < \min\limits_{e\in E_0} h(x,e),\\ 0, \text{иначе.}\end{cases} \quad (17)$$

Элементарный классификатор естественным образом может быть закодирован в форме набора $n+m$ хромосом, так чтобы каждый эталонный вектор соответствовал своей хромосоме.

Пример кодирования классификатора.
Параметры классификатора: n = 2, m=3, dim=8.
Группы: $E_0 = \{e_1, e_2\}$, $E_1 = \{e_3, e_4, e_5\}$.
Хромосомы: e_1 = 00010001, e_2 = 00111100, e_3 = 00000001, e_4 = 11111011, e_5 = 00011110.
Генетический код: 00010001 00111100 00000001 11111011 00011110.

На практике бывает полезным разбить полный вектор признаков x на вектора меньшей размерности, особенно в случае если вектор признаков x содержит в себе свойства различной природы. В нашем случае имеется два вектора pr, vl, для шаблонов цены и объема соответственно. После такого разбиения результирующий классификатор можно построить в виде произведения

$$d(x) = d_p(pr)d_v(vl), \quad (18)$$

где $x = \{pr, vl\}$.

Набор хромосом классификатора d включает в себя упорядоченные хромосомы каждого элементарного классификатора d_p и d_v. Комбинированный классификатор d рассматривается в качестве индивидуального классификатора при построении агрегированного классификатора d_{agr} методом бэггинга.

5. Алгоритм обучения классификатора

В этом разделе описывается генетический алгоритм, используемый для обучения элементарных классификаторов, а также способ агрегации получаемых классификаторов. Для обучения требуется три выборки: обучающая S, контрольная V и тестовая T.

Из исходной выборки S методом бутстрапа создаем набор подвыборок $\{S_j\}_{j=1}^{B}$. Применим к этим подвыборкам генетический алгоритм G^k, где k указывает на номер поколения в генетическом алгоритме. В результате мы получим B решающих правил $d_j^k = G^k(S_j)$ где $j = 1,...,B$. Далее выполняется «голосование» индивидуальных классификаторов:

$$\text{Margin}^k(x) = \frac{1}{B}\sum_{j=1}^{B} d_j^k(x). \quad (19)$$

В результате на каждом поколении k мы получаем агрегированный классификатор

$$d_{agr}^k(x)=\begin{cases}1, \text{если Margin}^k(x) > \theta^k\\ 0, \text{иначе.}\end{cases} \quad (20)$$

Порог θ^k выбирается так, чтобы $\overline{f}(d_{agr}^k, V)$ достигала максимума. В итоге, получаем набор агрегированных классификаторов $\{d_{agr}^k\}_{k=1}^{M}$. Результирующий классификатор берется как лучший среди $\{d_{agr}^k\}_{k=1}^{M}$ на контрольной выборке *V*.

При создании генетического алгоритма G^k существует большая степень свободы при выборе параметров алгоритма и операторов селекции, рекомбинации и мутации. Кратко опишем используемые операторы.

В качестве оператора селекции используется *ранговая селекция*. Выполняется она путем ранжирования классификаторов по мере увеличения функции эффективности, т.е. чем больше функция эффективности, тем больше ранг классификатора. Затем производится случайный отбор классификаторов с возвращением, с вероятностью их попадания в родительский пул пропорционально рангу классификатора.

Оператор скрещивания реализуется над родительским пулом следующим образом. Из родительского пула последовательно берется по два классификатора, и затем с высокой вероятностью p_c над ними выполняется скрещивания. Поскольку индивидуальный классификатор $d(x) = d_p(pr)d_v(vl)$, то над хромосомами элементарных классификаторов d_p и d_v скрещивание производится независимо. В результате скрещивания из исходных двух классификаторов родителей получаются два классификатора потомка, которые замещают собой родителей в родительском пуле. В итоге, получается популяция потомков размера исходного родительского пула. Вероятность p_c близка к 1 ($0.5 < p_c < 1$).

Рис. 6. Пример найденных классификатором моментов разворота цены на тестовой выборке

Оператор мутации выполняется над каждым классификатором из популяции потомков с малой вероятностью p_m близкой к 0 ($0< p_m <0.3$). Оператор мутации реализуется путем изменения одного случайного бита в одной из хромосом потомка. Мутация элементарных классификаторов d_p и d_v производится независимо.

6. Результаты

На Рис. 6 представлен пример найденных моментов разворота. На Рис. 7 представлена зависимость $\overline{f}(d,T)$ от количества индивидуальных классификаторов B в итоговом агрегированном классификаторе.

Параметры алгоритма:

Расстояние между барами t_0	5
Размер сетки для цены $w \times r_p$	6x32
Шаг дискретизации цены Δ_p	0.02
Размер сетки для объемов $w \times r_v$	6x32
Шаг дискретизации объема Δ_v	2
Размер популяции	50
Число итераций ГА	40
Параметр c в функции $\overline{f}(d,S)$	0.05
Параметры элементарных классификаторов	n = 2, m=2
Параметры генетического алгоритма	p_c=0.9 p_m=0.1

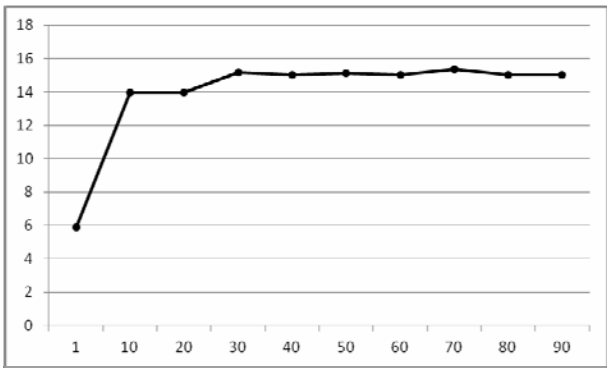

Рис. 7. Зависимость $\overline{f}$ на тестовой выборке от количества индивидуальных классификаторов усредненных по 5 запускам алгоритма

Заключение

Предложенный в работе алгоритм анализа финансовых временных рядов показал достаточно хорошие результаты при поиске моментов разворота в исследуемой модели. Стоит заметить, что классификатор не способен найти все моменты, однако, нахождение 70-80% моментов разворота вполне достижимо.

Предложенная схема классификации на основе простого отображения картины баров на решетку является упрощенной схемой. Для реальных рыночных ситуаций подобная система требует некоторого усложнения путем наложения дополнительных признаков. Применение описанного алгоритма для реальных финансовых рядов показало её состоятельность.

Для определенности перечислим основные достоинства и недостатки предложенного алгоритма. Высокая скорость работы готового классификатора является явным достоинством в задачах, где требуется принятие решения в реальном времени. С помощью побитовых операций можно быстро вычислять расстояние Хемминга между двумя бинарными векторами представленными набором бит.

Хорошая степень распараллеливания алгоритма позволяет достичь высокой скорости обучения. Простота реализации параллельного генетического алгоритма и независимость в создании индивидуальных классификаторов, все это дает возможность производить вычисления на многопроцессорных машинах и реализовывать алгоритм на вычислительном кластере. Основным недостатком алгоритма можно назвать большое число параметров, которые требуется настраивать для решения конкретной практической задачи.

Литература

1. Breiman, L. (1996). “Bagging predictors”. Machine Learning, 24, 123-140.
2. Breiman, L. (1998). “Arcing classifiers”. The Annals of Statistics, 26, 801-824.
3. Efron, B. Tibshirani, R.J. 1993, “An Introduction to the Bootstrap”. Chapman and Hall.
4. Kuncheva, L.: Combining pattern classifiers. John Wiley & Sons, Inc. (2004).
5. Lipman, R., An introduction to computing with neural nets// IEEE Acoustic, Speech and Signal Processing Magazine, 1987, no.2, L.4-22.
6. Devroye, L., Györfi, L., Lugosi, G.: A Probabilistic Theory of Pattern Recognition. Springer-Verlag New York, Inc. (1996).
7. Anirban Chakraborti, Ioane Muni Toke, Marco Patriarca & Frédéric Abergel (2011): Econophysics review: II. Agent-based models, Quantitative Finance, 11:7, 1013-1041.
8. Ladley, D., Zero Intelligence in Economics and Finance, The Knowledge Engineering Review, Vol. 00:0, 1–24.c 2004, Cambridge University Press.
9. Eiben, A. E., Smith, J. E., 2003, “Introduction to Evolutionary Computing”. Springer.
10. A. Lykov, S. Muzychka, K. L. Vaninsky, A multi-agent nonlinear markov model of the order book, http://arxiv.org/pdf/1208.3083.pdf.
11. http://thomsonreuters.com/.

Глекин Александр Олегович. Аспирант Института системного анализа РАН. Окончил Московский физико-технический институт (государственный университет) в 2010 году. Область научных интересов: машинное обучение, нейронные сети, распознавание образов, анализ данных, эконофизика. E-mail: graed@mail.ru