Введение в методы конкурентного предсказания

Ю. А. Калнишкан

Department of Computer Science
and Computer Learning Research Centre
Royal Holloway, University of London

2008

План доклада

- в докладе будет рассказано о
— задаче предсказания с использованием советов экспертов;
— соединяющем алгоритме, предложенном для её решения;
— универсальных методах предсказания, основанных на этом алгоритме.
- Литература:
— Prediction, learning, and games, Nicolò Cesa-Bianchi and Gábor Lugosi, Cambridge University Press, 2006
— Работы В. Вовка, С. Бусуттила, М. В. Вьюгина, В. В. Вьюгина, А. Гаммермана, Ф. Жданова, Ю. Калнишкана, К. Уоткинса и А. Чернова, доступные на сайте `vovk.net`

Содержание

Протокол

- мы наблюдаем элементы последовательности $\omega_1, \omega_2, \omega_3, \ldots \in \Omega$
- и выдаём предсказания $\gamma_1, \gamma_2, \gamma_3, \ldots \in \Gamma$
- протокол:

  FOR $t = 1, 2, \ldots$
  (1) $\mathfrak{A}$ выдаёт предсказание $\gamma_t \in \Gamma$
  (2) $\mathfrak{A}$ наблюдает исход $\omega_t \in \Omega$
  END FOR
- качество измеряется функцией потерь $\lambda(\omega, \gamma)$
  — *накопленные потери* (Loss) это сумма потерь за T шагов:

$$\mathrm{Loss}_{\mathfrak{A}}(\omega_1, \omega_2, \ldots, \omega_T) = \sum_{i=1}^{T} \lambda(\omega_i, \gamma_i)$$

Формализация

- *игра* $\mathfrak{G}$ это тройка $\langle \Omega, \Gamma, \lambda \rangle$, где
  — Ω это *пространство исходов*
  — Γ это *пространство предсказаний*
  — $\lambda : \Omega \times \Gamma \to [0, +\infty]$ это *функция потерь*
- под эту схему подпадает много разных протоколов взаимодействия со средой, не обязательно связанных с предсказанием
  — γ это действие, предпринятое в условиях неопределённости
  — ω это поведение среды
  — λ показывает, какой эффект дало наше действие

Двоичные игры

- в простейших *двоичных (бинарных) играх* $\Omega = \{0, 1\}$ и $\Gamma = [0, 1]$
- квадратичная игра: $\lambda(\omega, \gamma) = (\omega - \gamma)^2$
- абсолютная игра: $\lambda(\omega, \gamma) = |\omega - \gamma|$
- логарифмическая игра:

$$\lambda(\omega, \gamma) = \begin{cases} -\log_2(1 - \gamma) & \text{если} \quad \omega = 0 \\ -\log_2 \gamma & \text{если} \quad \omega = 1 \end{cases}$$

  — потери могут принимать значение $+\infty$
- простая предсказательная игра: $\Omega = \Gamma = \{0, 1\}$

$$\lambda(\omega, \gamma) = \begin{cases} 0 & \text{if} \quad \omega = \gamma \\ 1 & \text{if} \quad \omega \neq \gamma \end{cases}$$

Непрерывные игры

- у квадратичной и абсолютной игры есть *непрерывные* аналоги с $\Omega = \Gamma = [0, 1]$ и теми же функциями потерь
- логарифмическая игра имеет любопытные расширения, которые мы сейчас рассмотрим

Коверовская игра (1)

- рассмотрим N акций
- вектор $(\omega_t[1], \omega_t[2], \dots, \omega_t[N])$ описывает изменение рыночной коньюнктуры между моментами $t-1$ и t:
  — цена акции j меняется в $\omega_t[j]$ раз
- пусть $\gamma_t[j]$ это доля капитала, которую мы вкладываем в акцию j на j-м шаге
  — имеем $\gamma_t[j] \in [0,1]$ и $\sum_{j=1}^{K} \gamma_t[j] = 1$
- предположим, что наш капитал до шага t составлял W_{t-1}
  — в акцию j мы вкладываем $W_{t-1}\gamma_t[j]$
  — после изменения её цены, вложенная сумма превращается в $W_{t-1}\gamma_t[j]\omega_t[j]$
  — наш капитал меняется в $\sum_{j=1}^{N} \gamma_t[j]\omega_t[j] = \langle \gamma_t, \omega_t \rangle$ раз
- если в начальный момент наш капитал составлял $W_0 = 1$, то после T шагов имеем $W_T = \prod_{i=1}^{T} \langle \gamma_t, \omega_t \rangle$

Коверовская игра (2)

- коверовскую игру можно рассмотреть в нашей общей схеме
  — $\Omega = [0, +\infty]^N$ «положительный квадрант»
  — $\Gamma = \{(\gamma[1], \gamma[2], \dots, \gamma[N]) \in [0,1]^N \mid \sum_{j=1}^{N} \gamma[j] = 1\}$ симплекс
  — $\lambda(\omega, \gamma) = -\log_2 \langle \omega, \gamma \rangle$
- потери могут быть отрицательными; если это важно, и этот случай надо запретить, то вводят ограничение $\|\omega\|_\infty \le 1$ или меняют функцию потерь на $\log \|\omega\|_\infty - \log\langle \omega, \gamma \rangle$
- взяв логарифм, можем рассматривать потери в аддитивной модели; имеем

$$\mathrm{Loss}(\omega_1, \omega_2, \dots, \omega_T) = -\log_2 W_t$$

- капитал это экспонента от минус потерь

Игра Фройнда-Шапиро

- у меня есть N друзей, играющих на скачках (или в какую-то другую рискованную игру)
  — друг i несёт потери $\omega_t[i]$ на шаге t
  — предполагаем, что потери ограничены; после перенормировки можем считать, что $\omega_t[i] \in [0,1]$
- я раздаю свои деньги друзьям и прошу их сыграть за меня
  — если другу j на t-м шаге дана доля $\gamma_t[j]$, так что $\sum_{j=1}^{N} \gamma_t[j] = 1$, то мои потери будут выпуклой комбинацией $\lambda(\omega_t, \gamma_t) = \sum_{j=1}^{N} \omega_t[j]\gamma_t[j] = \langle \omega_j, \gamma_j \rangle$
- здесь $\Omega = [0,1]^N$, а Γ это симплекс, как в коверовской игре

Пример вне схемы: задача многорукого бандита

- заменим «друзей» из предыдущего примера на ручки игрального автомата
  — на каждом шаге я дёргаю за одну ручку и несу потери, соответствующие этой ручке
  — я не знаю, какие потери я бы понёс на другой ручке
  — обычно в этой игре разрешается рандомизация: я подбрасываю монетку, а от моих потерь берётся среднее
- это пример протокола с неполной информацией; он не покрывается нашей схемой непосредственно
- игру Фройнда-Шапиро можно рассматривать как вариант многорукого бандита с полной информацией и рандомизацией ($\gamma_t[j]$ можно рассматривать как вероятность выбора j-й ручки; я вычисляю эти вероятности и потом случайным образом выбираю ручку)

Протокол

- пусть N экспертов $\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_N$ предсказывают ту же самую последовательность
  — мы узнаём их предсказания до того, как выдаём наше
  — наша цель – предсказывать (почти) так же успешно, как лучший эксперт, в терминах накопленных потерь
- расширенный протокол:

(1) FOR $t = 1, 2, \ldots$
(2) $\mathfrak{M}$ считывает $\gamma_t^{(1)}, \gamma_t^{(2)}, \ldots, \gamma_t^{(N)} \in \Gamma$
(3) $\mathfrak{M}$ выдаёт предсказание $\gamma_t \in \Gamma$
(4) $\mathfrak{M}$ наблюдает исход $\omega_t \in \Omega$
(5) END FOR

Обсуждение

- мы хотим найти смешивающую стратегию, несущую потери, которые удовлетворяют условию

$$\mathrm{Loss}_{\mathfrak{M}} \leq f(\mathrm{Loss}_{\mathcal{E}_i})$$

где $\mathcal{E}_i$ это наиболее успешный на данный момент эксперт и $f(x)$ не намного больше x
- на экспертов не накладывается никаких ограничений
  — вычислительные возможности экспертов не ограничены
  — по сути «эксперт» это метафора последовательности предсказаний, подаваемых на вход протоколу
  — мы рассматриваем антагонистическую игру «предсказатель» против «природы и экспертов»
- эта проблема изучалась с конца 1980-х годов

Веса

- соединяющий алгоритм (Aggregating Algorithm) предложен В. Вовком в работах 1991 и 1998 гг.; это один из самых общих методов для решения задачи
- мы поддерживаем набор весов $(p_t^{(1)}, p_t^{(2)}, \ldots, p_t^{(N)})$ для экспертов
- на каждом шаге веса меняются в соответствии с потерями:

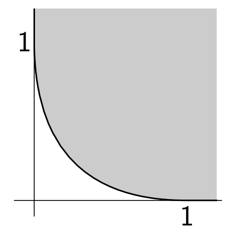
$$p_t^{(j)} = e^{-\eta\lambda(\omega_t, \gamma_t^{(j)})} p_{t-1}^{(j)}$$

— параметр $\eta \in (0, +\infty)$ называется *скоростью обучения*

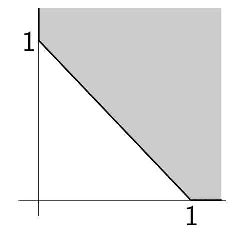
Обобщённые предсказания

- будем называть *обобщённым предсказанием* любую функцию на исходах, принимающую значения от 0 до $+\infty$ включительно, т.е. элемент множества $[0, +\infty]^\Omega$
  — для двоичных игр это множество можно отождествить с положительным квадрантом $[0, +\infty]^2$
- задание множества предсказаний и функции потерь выделяет в множестве предсказаний допустимые
  — предсказание γ задаёт функцию $\lambda(\cdot, \gamma) : \Omega \to [0, +\infty]$
- обобщённое предсказание, мажорирующее какое-либо предсказание, называется *суперпредсказанием*
  — иначе говоря, $g : \Omega \to [0, +\infty]$ является суперпредсказанием, если найдётся предсказание $\gamma \in \Gamma$, такое что $\lambda(\omega, \gamma) \le g(\omega)$ для всех ω

Примеры множеств суперпредсказаний (1)

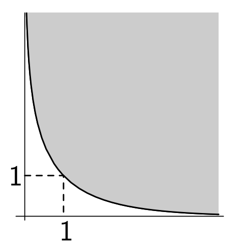


квадратичная игра
$\lambda(\omega, \gamma) = (\omega - \gamma)^2$



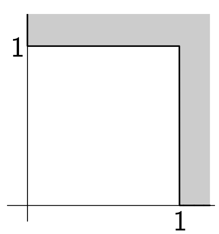
абсолютная игра
$\lambda(\omega, \gamma) = |\omega - \gamma|$

Примеры множеств суперпредсказаний (2)



логарифмическая игра

$$\lambda(\omega, \gamma) = \begin{cases} -\log_2(1-\gamma), & \omega = 0 \\ -\log_2 \gamma, & \omega = 1 \end{cases}$$



простая предсказательная игра

$$\lambda(\omega, \gamma) = \begin{cases} 0, & \omega = \gamma \\ 1, & \omega \ne \gamma \end{cases}$$

Смешивание

- получив предсказания экспертов, построим обобщённое предсказание g следующим образом

$$g(\omega) = -\log_\beta \sum_{j=1}^{N} \beta^{\lambda(\omega, \gamma_t^{(j)})} p_t^{(j)*}$$

  — где $\beta = e^{-\eta} \in (0, 1)$
  — $p_t^{(j)*}$ это веса, нормированные так, чтобы суммироваться к единице
- теперь нам надо произвести подстановку, выбрав предсказание γ, несущее потери не (намного) больше g
  — если g не суперпредсказание, нам придётся удовлетвориться неточной подстановкой

Константа $c(\beta)$

- пусть $\beta \in (0,1)$; обозначим через $c(\beta)$ наименьшее c со следующим свойством:
  — для любого набора предсказаний $\gamma^{(1)}, \gamma^{(2)}, \ldots, \gamma^{(N)} \in \Gamma$ и весов $p^{(1)}, p^{(2)}, \ldots, p^{(N)} \in [0,1]$, суммирующихся к единице, найдётся предсказание γ, такое что

$$\lambda(\omega, \gamma) \leq -c \log_\beta \sum_{j=1}^{N} \beta^{\lambda(\omega, \gamma^{(j)})} p^{(j)}$$

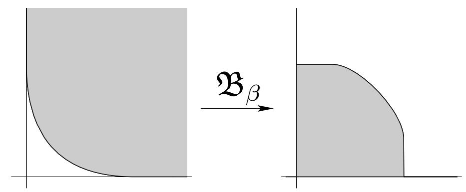
- если $c(\beta) = 1$ для какого-то β, то игра называется *смешиваемой (β-смешиваемой)*
- соединяющий алгоритм выдаёт γ_t, такое что $\lambda(\omega, \gamma_t) \leq c(\beta) g(\omega)$ для любого ω
  — иногда существует эффективная *функция подстановки*

Геометрическая интерпретация $c(\beta)$: смешиваемость

- рассмотрим преобразование $\mathfrak{B}_\beta$ множества обобщённых предсказаний, заданное так:

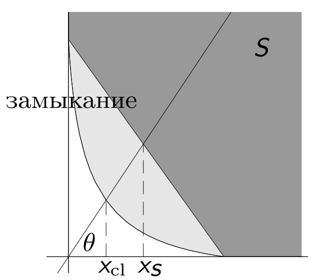
$$(\mathfrak{B}_\beta g)(\omega) = \beta^{g(\omega)}$$

- игра является β-смешиваемой тогда и только тогда, когда образ множества суперпредсказаний S под действием $\mathfrak{B}_\beta$ является выпуклым:



Геометрическая интерпретация $c(\beta)$: общий случай

- возьмём выпуклое замыкание образа $\mathfrak{B}_\beta(S)$
  — обозначим выпуклое замыкание буквой $\mathfrak{C}$
- вообще говоря $\mathfrak{C}(\mathfrak{B}_\beta(S))$ не совпадает с $\mathfrak{B}_\beta(S)$
  — $c(\beta)$ это «толщина» $\mathfrak{B}^{-1}(\mathfrak{C}(\mathfrak{B}_\beta(S))) \setminus S$



$c(\beta) = \max_\theta \frac{x_S}{x_{cl}}$

Смешиваемость конкретных игр (1)

- квадратичная игра смешиваема: $c(\beta) = 1$ для $\beta \geq 1$
- логарифмическая игра смешиваема: $c(\beta) = 1$ для $\beta \geq 1/2$
- абсолютная игра не смешиваема:

$$c(\beta) = \left(\ln \frac{1}{\beta}\right) \Big/ \left(2 \ln \frac{2}{1+\beta}\right)$$

- простая игра не смешиваема:

$$c(\beta) = \left(\ln \frac{1}{\beta}\right) \Big/ \left(\ln \frac{2}{1+\beta}\right)$$

— для смешиваемости необходима выпуклость множества суперпредсказаний

Смешиваемость конкретных игр (2)

- коверовская игра смешиваема: $c(\beta) = 1$ для $\beta \geq 1/2$ — и тогда не важно, что потери могут быть отрицательными
- игра Фройнда-Шапиро не смешиваема:

$$c(\beta) = \left(\ln \frac{1}{\beta}\right) \Big/ \left(N \ln \frac{N}{N + \beta - 1}\right)$$

Верхняя оценка на потери

- дадим экспертам одинаковый начальные веса $1/N$
- для любого эксперта $\mathcal{E}_i$ из произвольного набора из N экспертов, на любом шаге T и при любых исходах, если только стратегия $\mathfrak{M}$ пользуется соединяющим алгоритмом, то выполняется неравенство

$$\mathrm{Loss}_{\mathfrak{M}}(T) \leq c(\beta)\,\mathrm{Loss}_{\mathcal{E}_i}(T) + \frac{c(\beta)}{\ln(1/\beta)} \ln N$$

  — для смешиваемых игр мультипликативный коэффициент можно сделать равным 1
- в этой оценке ни мультипликативный ни аддитивный коэффициенты не растут со временем

Оптимальность верхней оценки

- предположим, для какого-либо алгоритма $\mathfrak{M}$ выполняется оценка

$$\mathrm{Loss}_{\mathfrak{M}}(T) \leq c\,\mathrm{Loss}_{\mathcal{E}_i}(T) + \alpha \ln N$$

  для любого эксперта $\mathcal{E}_i$ из произвольного набора из N экспертов, на любом шаге T и при любых исходах — тогда найдётся $\beta \in (0, 1)$, такое что $c(\beta) \leq c$ и $c(\beta)/\ln(1/\beta) \leq \alpha$
- константы в оценке, даваемой соединяющим алгоритмом, нельзя улучшить

Другие оценки

- оптимальность не запрещает построение оценок другого типа
- например, для абсолютной игры можно добиться оценки

$$\mathrm{Loss}_{\mathfrak{M}}(T) \leq \mathrm{Loss}_{\mathcal{E}_i}(T) + 2\sqrt{T \ln N}$$

  — здесь мультипликативный коэффициент равен единице, зато аддитивный растёт со временем как $\sqrt{T}$

Коверовская игра

- для коверовской игры получаем
$$\mathrm{Loss}_{\mathfrak{M}}(T) \le \mathrm{Loss}_{\mathcal{E}_i} + \frac{\ln N}{\ln 2}$$
- экспоненциируя и переходя от потерь к капиталу, получаем
$$W_{\mathfrak{M}}(T) \ge \frac{1}{N} W_{\mathcal{E}_i}(T)$$
- эта оценка совпадает с получаемой из тривиальных соображений:
— пусть N экспертов хотят советовать нам, как вкладывать деньги; поделим деньги между ними поровну и дадим каждому распоряжаться своей долей; тогда в каждый момент наш совокупный капитал будет не меньше, чем у каждого из них

Игра Фройнда-Шапиро

- обычно в игре Фройнда-Шапиро задача ставится так:
— как делить деньги между друзьями так, чтобы нести потери примерно как у самого успешного?
- рассмотрим N экспертов следующего вида:
— эксперт j даёт все деньги другу j, т.е. предсказывает $(0, \ldots, 0, 1, 0, \ldots, 0)$, где единица стоит на j-м месте
- соединяющий алгоритм может быть применён к задаче непосредственно
— поскольку игра не смешиваема, в оценке будет присутствовать мультипликативная константа; оценку нельзя однозначно оптимизировать
— но мультипликативная константа получается лучше, чем в оригинальной работе Фройнда и Шапиро [Вовк, 1998]

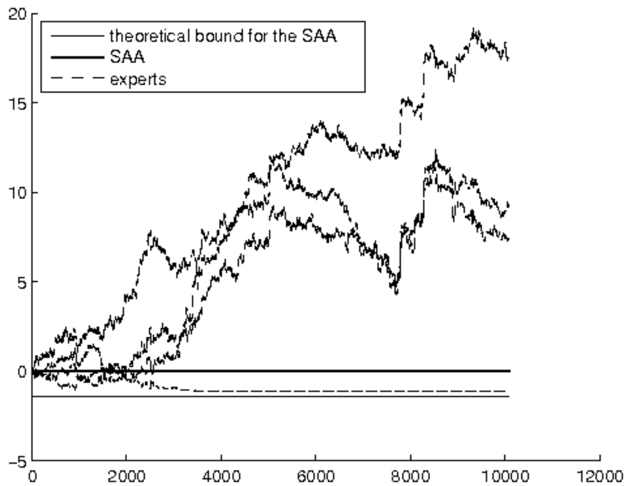
Тотализатор (1)

- букмекеры публикуют условия, на которых они принимают ставки
— можно считать, что они выдают предсказание на результат матча
— попробуем смешать предсказания разных букмекеров
- будем считать, что букмекер вычисляет распределение вероятностей и использует его, чтобы рассчитать условия приёма ставок
— в условия закладывается маржа
- восстановим (гипотетическое) распределение по ставкам
— будем считать, что букмекер выдаёт распределение вероятностей на исходы матча

Тотализатор (2)

- наша игра такова:
— у теннисного матча два исхода, победа одного или другого игрока; будем обозначать их векторами $(1, 0)$ и $(0, 1)$
— предсказание в нашей модели это распределение вероятностей, т.е. точка (p_1, p_2) из симплекса $p_1, p_2 \in [0, 1]$, $p_1 + p_2 = 1$
— будем мерить потери квадратичной функцией: $\lambda(\omega, \gamma) = \|\omega - \gamma\|^2$
— конечно, наша схема идеализирована
- можно показать, что эта игра (а также игры с 3-мя и более исходами) смешиваема
- ставки популярных букмекеров Великобритании доступны в Интернете; построим наши собственные предсказания

Разности потерь



графики $(\mathrm{Loss}_{\mathcal{E}_i}(T) - \mathrm{Loss}_{\mathfrak{M}}(T))$ по [Жданов и Вовк, 2008]

Капитал как экспонента потерь

- почему соединяющий алгоритм смешивает таким образом?
- одна из мотивировок – коверовская игра, рассмотренная выше
  — мы делим деньги между экспертами и даём им самостоятельно инвестировать
  — в случае произвольной игры «деньги» это $e^{-\eta \,\mathrm{Loss}}$
  — поскольку линейность коверовской игры у других игр отсутствует, подбор предсказания, соответствующего линейной комбинации, становится сложной задачей
  — появляется необходимость рассматривать коэффициент $c(\beta)$

Байесовской правило

- для логарифмической игры соединяющий алгоритм эквивалентен правилу Байеса

$$P(A_i \mid B) = \frac{P(A_i)P(B|A_i)}{\sum_i P(A_i)P(B|A_i)}$$

— $P(A_i)$ – нормированный вес эксперта i на шаге t
— $P(A_i|B)$ – нормированный вес эксперта i на шаге $t+1$, после того, как произошёл исход B
— $P(B|A_i)$ это предсказание исхода B на шаге t (т.е. $\gamma_t^{(i)}$ при исходе 1 и $1-\gamma_t^{(i)}$ при исходе 0)

Соединение больших семейств алгоритмов

- соединяющий алгоритм может применяться к счётным и континуальным семействам экспертов
- возьмём семейство методов предсказания с каким-либо параметром
  — каждое значение параметра можно считать «экспертом»
  — применив соединяющий алгоритм, получим универсальный непараметрический метод, несущий потери не намного хуже каждого исходного
- верхние оценки на потери носят особый характер
  — не накладывается никаких ограничений на механизм, порождающий данные
  — универсальный метод работает успешно, если только один из методов исходного класса работает успешно
  — оценки верны *при всех* возможных исходах

Постоянные балансируемые портфели

- рассмотрим стратегию инвестирования, основанную на поддержке постоянных пропорций между вложениями в разные акции
  — в коверовской игре такая стратегия соответствует выдаче одного и того же предсказания $(\gamma_t^{(1)}, \gamma_t^{(2)}, \ldots, \gamma_t^{(N)})$
  — стратегия не является статической; поскольку цены акций меняются, мы должны на каждом шаге балансировать портфель, чтобы поддерживать постоянные пропорции между вложениями
  — если мы следуем постоянному балансируемому портфелю b, то наш капитал растёт с начального значения W_0 до $W_0 \prod_{i=1}^T \langle b, \omega_i \rangle$
- для каждого такого портфеля b рассмотрим эксперта, который им пользуется

Построение универсального портфеля

- выдадим эксперту b долю $d\mu(b)$ исходного капитала, и он будет ей распоряжаться
  — за время T капитал эксперта b вырастет в $\prod_{i=1}^T \langle b, \omega_i \rangle$ раз
  — наш суммарный капитал вырастет в
$$\int_{S_{N-1}} \prod_{i=1}^{T} \langle b, \omega_i \rangle d\mu(b)$$
  раз, где S_{N-1} – симплекс в N-мерном пространстве
  — каждый раз мы инвестируем в соответствии с формулой
$$\frac{\int_{S_{N-1}} b \prod_{i=1}^T \langle b, \omega_i \rangle d\mu(b)}{\int_{S_{N-1}} \prod_{i=1}^T \langle b, \omega_i \rangle d\mu(b)}$$

Оценки

- при разных начальных распределениях $\mu(b)$ мы получаем разные оценки
- равномерное распределение на симплексе даёт неравенство
$$W_b(T) \le W(T)(T+1)^{N-1},$$
  где $W_b(T)$ – капитал произвольного постоянного балансируемого портфеля b, а $W(T)$ – наш капитал
- выбор распределения Дирихле позволяет понизить показатель степени:
$$W_b(T) \le W(T) 2 (T+1)^{(N-1)/2}$$
- в терминах потерь имеем добавочный член $O(N \ln T)$

Обсуждение

- коверовские универсальные портфели были построены независимо от соединяющего алгоритма в [Ковер и Ордентлих, 1996], но являются его частным случаем (для коверовской игры)
- для получения оценок на капитал не потребовалось никаких ограничений на изменение цен акций (вроде независимости или следования геометрическому броуновскому движению)
  — оценки остаются верными при любом поведении рынка

Задача регрессия на последовательностях

- внесём дополнение в наш протокол предсказания; пусть помимо исходов нам поступают *сигналы* x_i
- протокол:

  FOR $t = 1, 2, \ldots$
  (1) $\mathfrak{A}$ наблюдает сигнал x_t
  (2) $\mathfrak{A}$ выдаёт предсказание $\gamma_t \in \Gamma$
  (3) $\mathfrak{A}$ наблюдает исход $y_t \in \Omega$
  END FOR
- интуитивно, задача алгоритма состоит в том, чтобы выяснить зависимость y от x на лету; назовём эту задачу *регрессией на последовательностях* (on-line regression)
- в более традиционной постановке алгоритму сразу даётся набор пар (x, y) (обучающая выборка); эта постановка называется *пакетной* (batch)

Игра

- будем считать, что $y_i \in \Omega = [-Y, Y]$
  — ограниченность множества существенна
- разрешим предсказания из отрезка $\Gamma = [-Y, Y]$
  — расширение отрезка до всей числовой прямой ничего существенного не добавит; мы будем получать суперпредсказания, которые заведомо хуже предсказаний
- будем рассматривать квадратичные потери $\lambda(y, \gamma) = (y - \gamma)^2$
- это непрерывная квадратичная игра; она смешиваема
- будем считать, что $x_i \in \mathbb{R}^n$

Применение соединяющего алгоритма

- возьмём класс постоянных регрессоров
  — эксперт, пользующийся постоянным регрессором θ на каждом шаге выдаёт предсказание $\gamma_t = \theta' x_t$
- введём на $\mathbb{R}^n$ гауссовское распределение $\left(\frac{a\eta}{\pi}\right)^{n/2} e^{-\eta a \|\theta\|^2}$ (здесь $a > 0$ это произвольный параметр) и применим соединяющий алгоритм
  — для получения предсказаний и оценок необходимо подсчитать интеграл $\int e^{-Q(x)} dx$, где Q – положительно определённая квадратичная форма; его можно взять в явном виде, поэтому алгоритм получается простой
- назовём получившийся алгоритм *аггрегированной регрессией* (AAR) [Вовк 1998, 2001]
  — альтернативное название – предсказатель Вовка-Азури-Вармута

Оценка для линейного случая

- пусть $\|x_t\|_\infty \le X$ (т.е. все компоненты x_t ограничены X)
- тогда

$$\mathrm{Loss}_{\mathrm{AAR}}(T) \le \inf_\theta(\mathrm{Loss}(T) + a\|\theta\|^2) + nY^2 \ln\left(\frac{TX^2}{2} + 1\right)$$

- можно показать, что эта оценка точна

Дуальная форма

- сам алгоритм может быть выписан в дуальной форме:

$$\gamma_T = \widetilde{Y}'_T(aI + \widetilde{K}_T)^{-1}\widetilde{k}_T$$

где
— $\widetilde{Y}'_T = (y_1, y_2, \ldots, y_{T-1}, 0)$
— $\widetilde{K}_T = (\langle x_i, x_j\rangle)_{i,j=1}^T$
— $\widetilde{k}_T = (\langle x_i, x_T\rangle)_{i=1}^T$

- эта запись наводит на мысль о родстве с *гребневой регрессией* (ridge regression)

Гребневая регрессия

- дуальная форма гребневой регрессии:

$$\gamma_T = Y_T(aI + K_T)^{-1}k_T$$

где
— $Y'_T = (y_1, y_2, \ldots, y_{T-1})$
— $K_T = (\langle x_i, x_j\rangle)_{i,j=1}^{T-1}$
— $k_T = (\langle x_i, x_T\rangle)_{i=1}^{T-1}$

- определение гребневой регрессии:
— пусть дана обучающая выборка $(x_1, y_1), \ldots, (x_{T-1}, y_{T-1})$
— выберем регрессор θ, минимизирующий сумму

$$\sum_{i=1}^{T-1}(\theta' x_i - y_i)^2 + a\|\theta\|^2$$

Сравнение двух типов регрессии (1)

- аггрегированная регрессия – это гребневая регрессия, добавляющая к обучающей выборке фиктивный пример $(x_T, 0)$, где x_T это сигнал, для которого надо получить предсказание
- можно сказать, что аггрегированная регрессия пользуется на x_T регрессором θ, минимизирующим сумму

$$\sum_{i=1}^{T-1}(\theta' x_i - y_i)^2 + (\theta' x_t)^2 + a\|\theta\|^2$$

— предсказание гребневой регрессии прижимается к 0

Сравнение двух типов регрессии (2)

- гребневая регрессия – байесовский метод; её предсказание минимизирует ожидаемые потери в предположении нормального распределения
- для аггрегированной регрессии есть верхние оценки на потери в худшем случае
— они выполняются при любых исходах

Кернелизация

- аггрегированная регрессия может быть кернелизована
— скалярные произведения $\langle x_i, x_j \rangle$ в дуальной форме заменяются на значения кернела $\mathcal{K}(x_i, x_j)$
- кернел можно определить как скалярное произведение в гильбертовом пространстве:

$$\mathcal{K}(x_i, x_j) = \langle (\Phi(x_i), \Phi(x_j) \rangle$$

— где Φ отображает x_i в гильбертово пространство H
— само отображение знать обычно не нужно; достаточно уметь вычислять кернел
- кернел задаёт функциональное пространство RKHS (reproducing kernel Hilbert space)
— грубо говоря, оно состоит из функций вида $\sum a_i \mathcal{K}(x_i, \cdot)$ и их пределов

Оценки в кернельном случае

- пусть $\mathcal{K}(x_i, x_j) \le c_{\mathcal{F}}$
- если T и верхняя оценка d на норму f из RKHS известны заранее, можно подобрать a так, чтобы выполнялось неравенство

$$\mathrm{Loss}_{\mathrm{KAAR}} \le \mathrm{Loss}_f(T) + 2Yc_{\mathcal{F}}d\sqrt{T}$$

- применением соединяющего алгоритма можно устранить параметр a совсем и получить верхнюю оценку типа

$$\mathrm{Loss}_{\mathrm{KAAR}} \le \mathrm{Loss}_f(T) + O(\sqrt{T})$$

— но соответствующий алгоритм в явном виде не выписывается

Практические применения

- численные эксперименты на модельных базах данных показывают, что дополнительная регуляризация, вносимая аггрегированной регрессией (по сравнению с гребневой), по-видимому является слишком сильной
— были сделанны попытки ослабить её, например, вместо

$$\sum_{i=1}^{T-1} (\theta' x_i - y_i)^2 + (\theta' x_t)^2 + a\|\theta\|^2 \to \min_{\theta}$$

рассмотреть

$$\sum_{i=1}^{T-1} (\theta' x_i - y_i)^2 + \beta(\theta' x_t)^2 + a\|\theta\|^2 \to \min_{\theta}$$

с $\beta < 1$ [Бусуттил и Калнишкан, 2007]

Меняющиеся регрессоры

- зависимость y от x может медленно меняться со временем
- рассмотрим класс регрессоров, которые меняются со временем
  — меняющийся регрессор это последовательность $\theta_1, \theta_2, \ldots$, и на шаге T он предсказывает $\gamma_T = (\theta_1 + \theta_2 + \ldots_T)' x_T$
- на таких регрессорах тоже можно ввести нормальное распределение и применить соединяющий алгоритм [Бусуттил и Калнишкан, 2007]
- получаемый алгоритм кернелизуется

Кернельная форма

$$\gamma_T = \widetilde{Y}'_T (aI + \widehat{K}_T)^{-1} \widehat{k}_t$$

где $\widetilde{Y}'_T = (y_1, y_2, \ldots, y_{T-1}, 0)$

$$\widehat{K}_T = \begin{pmatrix} \mathcal{K}(x_1, x_1) & \mathcal{K}(x_1, x_2) & \mathcal{K}(x_1, x_3) & \ldots & \mathcal{K}(x_1, x_T) \\ \mathcal{K}(x_2, x_1) & 2\mathcal{K}(x_2, x_2) & 2\mathcal{K}(x_2, x_3) & \ldots & 2\mathcal{K}(x_2, x_T) \\ \mathcal{K}(x_3, x_1) & 2\mathcal{K}(x_3, x_2) & 3\mathcal{K}(x_3, x_3) & \ldots & 3\mathcal{K}(x_3, x_T) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{K}(x_T, x_1) & 2\mathcal{K}(x_T, x_2) & 3\mathcal{K}(x_T, x_3) & \ldots & T\mathcal{K}(x_T, x_T) \end{pmatrix}$$

$$\widehat{k}_T = (\mathcal{K}(x_T, x_1), 2\mathcal{K}(x_T, x_2), 3\mathcal{K}(x_T, x_3), \ldots, T\mathcal{K}(x_T, x_T))'$$

— более старые сигналы входят с меньшими весами

Верхняя оценка на потери

- если T известно заранее и для $f_1, f_2, \ldots, f_t$ из RKHS имеем оценку

  $$\|f_i\| \le \frac{d}{T^{0.5+\varepsilon}}$$

  то получаем оценку на потери

  $$\mathrm{Loss}_{\mathrm{KAARCh}} \le \mathrm{Loss}_{f_1, f_2, \ldots, f_T}(T) + O(T^{1-\varepsilon})$$
- оценка не охватывает все меняющиеся регрессоры, а только медленно меняющиеся

Применения

- алгоритм применялся для предсказания неявной волатльности (implied volatility)
- неявная волатильность представляет собой функцию от цены исполнения опциона $\sigma(X)$
  — её часто называют улыбкой волатильности
  — она меняется по мере приближения к сроку исполнения опциона
- исход y_i соответствует цене сделки (или цене закрытия); сигнал X_i включает в себя цену исполнения, текущую цену актива и т.д.