

Санкт-Петербургский государственный университет
Математико-механический факультет
Кафедра статистического моделирования

Анастасия Миллер

**«УСТРАНЕНИЕ ЭКСПОНЕНЦИАЛЬНОЙ
СЛОЖНОСТИ ОЦЕНКИ СТОИМОСТИ
БЕРМУДСКОГО ОПЦИОНА»**

Курсовая работа

Научный руководитель
д.ф.-м.н. Ермаков С.М.

Рецензент

Санкт-Петербург
September 22, 2014г.

Содержание

1	Вступление	4
2	Метод случайных деревьев	4
2.1	Построение дерева и оценок	4
2.1.1	Оценка сверху	5
2.1.2	Оценка снизу	6
3	Устранение экспоненциальной сложности	7
3.1	Анализ распределения состояний с помощью гистограммы	7
3.2	Кластеризация состояний	8
4	Выбор k	9

1 Вступление

Метод оценки американских опционов с конечным числом дат погашения, основанный на моделировании дерева событий (метод случайных деревьев), был предложен в ещё в 2004 году. Этот метод моделирует изменение состояния базового актива через случайные деревья, разветвляющиеся в каждой из возможных дат раннего погашения опциона. При анализе деревьев могут быть получены две оценки: смещённая вверх и смещённая вниз, являющиеся асимптотически несмещёнными и дающие доверительный интервал для истинной цены опциона.

Один из основных недостатков алгоритма – его экспоненциальная сложность. Здесь же предлагается несколько подходов, которые заменят экспоненциальную сложность полиномиальной с одновременным увеличением «случайности» алгоритма.

2 Метод случайных деревьев

Обозначения и умолчания Будем строить модель на примере американского опциона с конечным числом дат погашения t_1, \dots, t_m (также называемого бермудским опционом). Мы также сузим класс решаемых нами задач до тех, в которых вся необходимая информация об активе, на который выписан рассматриваемый опцион, может быть представлена в виде Марковского процесса $X(t), t \in \{t_i\}_{i=1}^m$ со значениями в \mathbb{R}^d . Для уменьшения объёма текста будем обозначать $X(t_i) \equiv X_i$. Положим также $h_i(x)$ – размер выплаты по опциону в момент t_i при том, что $x = X_i$ и опцион не был исполнен до этого, $V_i(x)$ – стоимость опциона в момент t_i при том, что $x = X_i$.

Нетрудно видеть, что

$$V_m(x) = h_m(x) \quad (1)$$

$$V_{i-1}(x) = \max \{h_{i-1}(x), \mathbb{E}[V_i(X_i) | X_{i-1} = x]\} \quad (2)$$

– на каждом шаге мы выбираем наиболее выгодное решение. Здесь нас интересует значение $V_0(X_0)$.

2.1 Построение дерева и оценок

Вместо того, чтобы строить оценку, каким-либо образом стремящуюся к требуемому нами значению, мы построим две оценочные функции, оценивающие V_n сверху и снизу. Пусть $\hat{V}_n(b)$ и $\hat{v}_n(b)$ – такие оценки, зависящие от некоторого параметра b .

Метод случайного дерева основан на моделировании цепи X_0, X_1, \dots, X_n . Зафиксируем параметр ветвления b . Из исходного состояния X_0 смоделируем b независимых следующих состояний $X_1^1, X_1^2, \dots, X_1^b$, все с условием X_1 . Для каждого X_1^i снова смоделируем b независимых последующих состояний $X_2^{i1}, \dots, X_2^{ib}$. На m -ом шаге будем иметь b^m состояний, и это и есть источник основного недостатка этого метода – его экспоненциальной алгоритмической сложности.

2.1.1 Оценка сверху

Определим $\hat{V}_i^{j_1, j_2, \dots, j_i}$, вдохновляясь 1. В последних вершинах (листьях) дерева зададим

$$\hat{V}_m^{j_1 \dots j_m} = h_m(X_m^{j_1 \dots j_m}) \quad (3)$$

Идя вверх по дереву, зададим

$$\hat{V}_i^{j_1 \dots j_i} = \max \left\{ h_i(X_i^{j_1 \dots j_i}), \frac{1}{b} \sum_{j=1}^b \hat{V}_{i+1}^{j_1 \dots j_i j} \right\} \quad (4)$$

С помощью индукции можно доказать, что наша оценка уклоняется вверх в каждом узле

Теорема 1. $\forall i \in 1 : n$

$$\mathbb{E} \left[\hat{V}_i^{j_1 \dots j_i} | X_i^{j_1 \dots j_i} \right] \geq V_i(X_i^{j_1 \dots j_i})$$

Доказательство. В листьях дерева неравенство выполняется как равенство по определению.

Докажем, что если утверждение теоремы выполняется на $i+1$ шаге, то оно выполняется и на i . По определению

$$\mathbb{E} \left[\hat{V}_i^{j_1 \dots j_i} | X_i^{j_1 \dots j_i} \right] = \mathbb{E} \left[\max \left\{ h_i(X_i^{j_1 \dots j_i}), \frac{1}{b} \sum_{j=1}^b \hat{V}_{i+1}^{j_1 \dots j_i j} \right\} | X_i^{j_1 \dots j_i} \right]$$

с помощью неравенства Йенсена ($\varphi(\mathbb{E}[X]) \leq \mathbb{E}[\varphi(X)]$) это можно оценить

$$\mathbb{E} \left[\max \left\{ h_i(X_i^{j_1 \dots j_i}), \frac{1}{b} \sum_{j=1}^b \hat{V}_{i+1}^{j_1 \dots j_i j} \right\} | X_i^{j_1 \dots j_i} \right] \geq \max \left\{ h_i(X_i^{j_1 \dots j_i}), \mathbb{E} \left[\frac{1}{b} \sum_{j=1}^b \hat{V}_{i+1}^{j_1 \dots j_i j} | X_i^{j_1 \dots j_i} \right] \right\}$$

в силу того, что $\forall j \in 1 : b$ $X_{i+1}^{j_1 \dots j_i j}$ - независимые одинаково распределённые случайные величины (и их математическое ожидание одинаково), $\mathbb{E} \left[\frac{1}{b} \sum_{j=1}^b \hat{V}_{i+1}^{j_1 \dots j_i j} \right] = \frac{1}{b} \sum_{j=1}^b \mathbb{E} \hat{V}_{i+1}^{j_1 \dots j_i j} = \mathbb{E} \hat{V}_{i+1}^{j_1 \dots j_i 1}$, а в силу индукционного предположения

$$\max \left\{ h_i(X_i^{j_1 \dots j_i}), \mathbb{E} \left[\hat{V}_{i+1}^{j_1 \dots j_i 1} | X_i^{j_1 \dots j_i} \right] \right\} \geq \max \left\{ h_i(X_i^{j_1 \dots j_i}), V_i(X_i^{j_1 \dots j_i}) \right\}$$

Таким образом, $\mathbb{E} \left[\hat{V}_i^{j_1 \dots j_i} | X_i^{j_1 \dots j_i} \right] \geq \max \left\{ h_i(X_i^{j_1 \dots j_i}), V_i(X_i^{j_1 \dots j_i}) \right\}$ □

Мы также доказываем, что $\hat{V}_i^{j_1 \dots j_i}$ сходится по вероятности к $V_i(X_i^{j_1 \dots j_i})$ при $b \rightarrow \infty$. В листьях дерева это очевидно ($\hat{V}_m^{j_1 \dots j_m} = h_m(X_m^{j_1 \dots j_m})$ по определению), на $i-1$ шаге цена удержания опциона $\frac{1}{b} \sum_{j=1}^b \hat{V}_{i+1}^{j_1 \dots j_i j}$ является средним арифметическим независимых одинаково распределённых случайных величин и сходится по закону больших чисел.

Сходимость распространяется и на саму оценку в силу непрерывности операции взятия максимума. Используя тот факт, что $\forall a, c_1, c_2 \in \mathbb{R} \mid \max(a, c_1) - \max(a, c_2) \mid \leqslant |c_1 - c_2|$, мы получаем

$$\left| \hat{V}_i^{j_1 \dots j_i} - V_i(X_i^{j_1 \dots j_i}) \right| \leqslant \frac{1}{b} \sum_{j=1}^b \left| \hat{V}_{i+1}^{j_1 \dots j_i j} - \mathbb{E}[V_{i+1}(X_{i+1}^{j_1 \dots j_i j}) \mid X_{i+1}^{j_1 \dots j_i}] \right|$$

что позволяет нам вывести из сходимости на $i+1$ шаге сходимость на i шаге. Подробнее в [2]

Более того, асимптотически наша оценка оказывается не сдвинутой вверх, т.е. $\mathbb{E} \hat{V}_0 \rightarrow V_0(X_0)$

2.1.2 Оценка снизу

Значения оценки сверху в каждый момент времени – это выбор максимума из стоимости опциона при его немедленном исполнении и математического ожидания стоимости удержания опциона. Но стоимость удержания опциона рассчитывается, исходя из дочерних узлов дерева состояний актива, то есть оценка сверху рассчитывается, опираясь на информацию о будущем. Чтобы убрать ошибку, связанную с этим, нам необходимо отделить механизм принятия решения о исполнении/удержании опциона от значений, полученных после принятия решения об удержании опциона.

По сути, нам нужно оценить $\max\{a, \mathbb{E}Y\}$ с помощью b независимых одинаково распределённых реализаций случайной величины Y для некоторой константы a и случайной величины Y . Оценка $\max\{a, \bar{Y}\}$ (где \bar{Y} – среднее значение выборки) является оценкой сверху, так как $\mathbb{E} \max\{a, \bar{Y}\} \geqslant \max\{a, \mathbb{E}\bar{Y}\} = \max\{a, \mathbb{E}Y\}$, что мы и использовали в построении нашей оценки сверху.

Разделим наше множество реализаций $\{Y_i\}_{i=1}^b$ случайной величины Y на два независимых подмножества и вычислим их средние значения \bar{Y}_1 и \bar{Y}_2 . Если положить

$$\hat{v} = \begin{cases} a, & \text{если } \bar{Y}_1 \leqslant a \\ \bar{Y}_2, & \text{иначе} \end{cases} \quad (5)$$

мы отделим процесс принятия решения о исполнении/удержании опциона от оценки его стоимости (за решение будет отвечать \bar{Y}_1 , за оценку – \bar{Y}_2). При этом оценка \hat{v} является оценкой снизу:

$$\mathbb{E} \hat{v} = \mathbb{P}(\bar{Y}_1 \leqslant a) a + (1 - \mathbb{P}(\bar{Y}_1 \leqslant a)) \mathbb{E}Y \leqslant \max\{a, \mathbb{E}Y\} \quad (6)$$

Нижняя оценка по [4] Пусть «отвечающим за принятие решения» подмножеством будут все реализации, кроме одной, а «оценивающее» множество будет состоять из одной оставшейся реализации. Возьмём математическое ожидание этой величины, т.е. положим в листьях дерева значение оценки

$$\hat{v}_m^{j_1 j_2 \dots j_m} = h(X_m^{j_1 j_2 \dots j_m}) \quad (7)$$

а для промежуточных узлов определим

$$\hat{v}_{ik}^{j_1 j_2 \dots j_i} = \begin{cases} h(X_i^{j_1 j_2 \dots j_i}), & \text{если } \frac{1}{b-1} \sum_{j=1, j \neq k}^b \hat{v}_{i+1}^{j_1 j_2 \dots j_i j} \leq h(X_i^{j_1 j_2 \dots j_i}) \\ \hat{v}_{i+1}^{j_1 j_2 \dots j_i k}, & \text{иначе} \end{cases} \quad (8)$$

и оценку положим равной

$$\hat{v}_i^{j_1 j_2 \dots j_i} = \frac{1}{b} \sum_{k=1}^b \hat{v}_{ik}^{j_1 j_2 \dots j_i}$$

3 Устранение экспоненциальной сложности

Начиная с некоторого момента t_k , когда общее число состояний достигнет некоторого n , мы перестанем генерировать дочерние вершины ко всем состояниям. В следующий момент времени, t_{k+1} , мы будем иметь всё так же n состояний, а не bn . Этого можно достичь, если генерировать дочерние состояния не ко всем вершинам, а только к некоторым. К каким?

3.1 Анализ распределения состояний с помощью гистограммы

В том случае, когда состояние актива S является числом в \mathbb{R}^1 , в качестве параметра X , распределение которого нас интересует, можно использовать само S , иначе можно использовать $h(S)$.

Деля интервал $[\min_{i \in 1:n} X_i; \max_{i \in 1:n} X_i + \frac{1}{n})$ на k равных частей $[a_{k-1}, a_k)$, где $a_0 = \min_{i \in 1:n} X_i$, $a_k = \max_{i \in 1:n} X_i$, мы можем определить частоты

$$f_k = \frac{1}{n} \# \{X_i | X_i \in [a_{k-1}, a_k)\}$$

попадания событий в различные части отрезка. Из состояний, сгруппированных на отрезке $[a_{k-1}, a_k)$, мы также можем создать некоторый «средний арифметический» вектор, координаты которого будут являться средним арифметическим координат всех состояний, оказавшихся на данном отрезке, и уже для этого нового среднего состояния – представителя отрезка – генерировать дочерние вершины в количестве $n \cdot f_k$. Для всех состояний, оказавшихся в этом отрезке, дочерними вершинами будут являться все вершины, полученные от их представителя.

Таким образом, количество рассматриваемых состояний не увеличится. С другой стороны, этот метод предполагает хранение в памяти всего дерева, а не только непосредственно обчитываемой ветки, как это предполагалось в исходной работе [2].

Гистограммный анализ «среза» распределения проходит за $O(n)$ времени и за $O(n)$ по памяти, что приводит нас к общему времени работы $O(mn)$ и общей затраченной памяти $O(mn)$, где m – число дат погашения опциона, включая первую и последнюю.

3.2 Кластеризация состояний

Для выделения родителей будущего поколения событий можно использовать не гистограммный подход, а кластеризацию существующего поколения. Так как состояния являются векторами в \mathbb{R}^d , в качестве метрики можно взять, например, «улучшенную» евклидову метрику в \mathbb{R}^d :

$$\mu(S_i, S_j) = \sqrt{\sum_{k=1}^d \frac{(s_k^i - s_k^j)^2}{c_k}},$$

где $c_k = |\max_{i \in 1:n} s_k^i - \min_{i \in 1:n} s_k^i|$, т.е. масштабирующий множитель, уравнивающий влияние различных компонент состояний на итоговое расстояние между ними.

Кластеризация может быть проведена по любому из известных алгоритмов, приведём один из наиболее простых и популярных алгоритмов кластеризации применительно к нашей задаче – алгоритм k -средних.

```
// назначаем центроидами кластеров случайно выбранные  $S_k$ 
for  $j \in 1 : k$  do
    Get( $\alpha$ )
     $C_j = S_{\lceil n\alpha \rceil}, \forall i \in 1 : j - 1, C_i \neq C_j$ 
end
for  $j \in 1 : n$  do
    // центроидом для каждого состояния полагаем тот из центроидов,
    // который ближе всего к данному состоянию
     $S_j.\text{centroid} = \text{argmin}(i \in 1 : k, \mu(S_j, C_i))$ 
end
changed = true
repeat
    for  $j \in 1 : k$  do
        centroid =  $\{i \in 1 : n | S_i.\text{centroid} = C_j\}$ 
         $C_j = \frac{1}{\#(\text{centroid})} \sum_{i \in \text{centroid}} S_i$ 
    end
    changed = false
    for  $j \in 1 : n$  do
        // пересчитываем принадлежность состояний центроидам
        oldcentroid =  $S_j.\text{centroid}$ 
         $S_j.\text{centroid} = \text{argmin}(i \in 1 : k, \mu(S_j, C_i))$ 
        changed = (oldcentroid  $\neq$   $S_j.\text{centroid}$ )
    end
until changed = true
```

В этом случае набор центроидов $\{C_j\}_{j=1}^k$ можно использовать в качестве родителей для следующего поколения состояний.

Известные недостатки алгоритма k -средних, такие как сложность $O(2^n)$ в худшем случае, зависимость результатов от начального выбора центроидов (которая, впрочем, может быть частично устранена при модификации алгоритма, например, в версии k -means++, описанной в [1]), влияют на итоговую сложность алгоритма. Как и в случае гистограммного анализа, здесь потребуется $O(mn)$ памяти в силу необходимости хранить все когда-либо сгенерированные узлы, но затраты на анализ среза будут равны $O(n)$ в лучшем случае и $O(2^n)$ в худшем, что ведёт к сложности $O(m \cdot 2^n)$ для всего процесса оценки в худшем случае.

Возможным преимуществом перед гистограммным анализом может оказаться более «честный» выбор представителя группы, не зависящий от того, как именно группа состояний распределена относительно границ отрезков, на которые мы делим область распределения.

4 Выбор k

Выбор количества родителей для будущего поколения – отдельный вопрос. Скорее всего, количество родителей будет непосредственно влиять на величину доверительного интервала результата.

Список литературы

- [1] David Arthur и Sergei Vassilvitskii. «k-means++: The Advantages of Careful Seeding». В: *SODA*. 2007. URL: <http://theory.stanford.edu/~sergei/papers/kMeansPP-soda.pdf>.
- [2] М. Broadie и Р. Glasserman. «Pricing American-style securities by simulation». Английский. В: *Journal of Economic Dynamics and Control* 21 (1997), с. 1323–1352.
- [3] Mark Broadie, Paul Glasserman и Gautam Jain. «Enhanced Monte Carlo estimates for american option prices». В: *Journal of Derivatives* 5.1 (Fall) (1997), с. 25–44.
- [4] Paul Glasserman. *Monte Carlo Methods in Financial Engineering*. Английский. Springer, 2004.