

## アップサイドチャンスポートフォリオ最適化 に関する数値的考察

CHEN, YaoYao / 陳, 瑶瑶

---

(発行年 / Year)

2013-03-24

(学位授与年月日 / Date of Granted)

2013-03-24

(学位名 / Degree Name)

修士(工学)

(学位授与機関 / Degree Grantor)

法政大学 (Hosei University)

2012 年度修士論文

# アップサイドチャンスポートフォリオ 最適化に関する数値的考察



法政大学工学研究科  
システム工学専攻

11R6214 陳 瑶瑶

指導教員 浦谷 規 教授

THE 2012 MASTER'S THESIS

NUMERICAL EXPERIMENTS WITH  
RESPECT TO UP-SIDE CHANCE  
PORTFOLIO OPTIMIZATION



DEPARTMENT OF SYSTEM ENGINEERING  
GRADUATE SCHOOL OF ENGINEERING  
HOSEI UNIVERSITY

11R6214 YAOYAO CHEN

SUPERVISOR PROF. TADASHI URATANI

## 目次

1	はじめに	3
2	モデル設定	3
1	ファクターモデルの設定	3
2	株価モデルの設定	3
3	安全資産モデルの設定	3
4	株と安全資産のポートフォリオ	3
5	基準値 (benchmark) のモデル設定	4
6	ポートフォリオと基準値の比率	4
7	投資額の成長率	4
8	アップサイドチャンスの問題設定	4
3	アップサイドチャンスの問題設定	4
1	Pham[1] の主結果	5
2	Pham[1] の主結果についての証明	6
4	ファクターモデルの場合分け	9
1	ファクターモデルが定数	9
2	ファクターモデルが Vasicek interest rate model	9
5	数値実験の方法	10
1	Black-Scholes model のオイラ-丸山近似	10
2	Vasicek model のオイラ-丸山近似	10
3	数値実験のモデル係数について	11
6	数値実験の結果と考察	11
1	株価が Black-Scholes Model の数値実験の結果	11
2	株価が Black-Scholes Model の数値実験の結果についての考察	16
3	Black-Scholes Model の理論値と実験値の誤差	16
4	ファクターモデルが Vasicek Model の数値実験の結果	19
5	ファクターモデルが Vasicek Model の数値実験の結果についての考察	23
7	付録	23
1	Black-Scholes model の数値実験プログラム	23
2	Vasicek interest rate model の数値実験プログラム	27
8	謝辞	33

## 概要

本研究は大偏差原理を用いたアップサイドチャンスポートフォリオ最適化の数値実験を行い、目標の成長率とそれを達成する確率の関係を考察する。Pham[1]の理論に基づいて数値実験を行う。数値実験では証券に影響を与えるファクターに2つの場合を考える。第1はファクターが定数のときで、株価はブラックショールズと同じになる。このとき、ポートフォリオが目標成長率を達成する確率を具体的に計算することができる。その理論値とシミュレーションで得た数値を比較し、誤差がどのくらい生じるかを把握する。第2は、ファクターにバシチェックモデルを仮定し、実験を行う。バシチェックモデルでは理論的な解が得られないので、数値シミュレーションにより、目標成長率とそれを達成する確率の関係を明らかにし、特徴を捉える。

## 概要

In this paper we consider up-side chance portfolio optimization which is obtained by using large deviation theory. Its theoretical aspects were studied in Pham[1]. We numerically experiment relationships between a target growth rate and the probability of attainability. We adopt factor model for security prices. We consider two cases: the first case is that the factor is a constant, namely the security price becomes log-Normal process. In this case, we are able to calculate the probability that the portfolio achieves the target growth rate exactly. By comparing the numerical results with the theoretical ones, we examine the accuracy of our numerical results. The second case is the Vasicek model of factor. In this case, we do not obtain the analytical solution of the probability. Therefore we clarify relationships between the target growth rate and the probability to achieve it through numerical experiments.

## 1. はじめに

大偏差原理を用い、アップサイドチャンスの数値実験を行うことが本研究の目的である。大偏差原理とは、簡潔にいうと分布の裾を調べ、平均から大きく外れたところの点を研究する原理である。金融市場からいうと、頻繁に起こらない市場現象を指している。例えば、アップサイドチャンスの例だと、設定した目標値を超えた確率を調べ、目標値よりどのくらい上回るのか具体的に調べる際に、大偏差原理を用いた。本研究は大偏差原理を用いているため、投資期間を長期として、短期だと、H.Pham の論文が適用されにくいである。その故、数値実験の際に、実際の市場の状況を踏まえて、期間を 5 年から 30 年だと設定した。

## 2. モデル設定

### (1) ファクターモデルの設定

$Y$  は経済的なファクター、市場や証券に影響を与える要素で、金利、為替などとし、次の確率微分方程式に従うとする; $B$  が標準ブラウン運動である。

$$dY_t = -k(Y_t - \bar{y})dt + vdB_t, Y_0 = y_0 \quad (1)$$

但し、 $k > 0, \bar{y}, y_0 \in R, v \neq 0$  とする。 $B$  は標準ブラウン運動である。以下では一般性を失わず  $\bar{y} = 0$  とする。

### (2) 株価モデルの設定

株価の過程  $S$  は次の確率微分方程式に従うとする;

$$\frac{dS_t}{S_t} = \mu(Y_t)dt + \sigma dW_t \quad (2)$$

$W$  を標準ブラウン運動とする。 $W$  と  $B$  は  $d \langle W, B \rangle = \rho dt, \rho \in [-1, 1]$  とする。ボラティリティ  $\sigma$  を 0 より大きい定数とする。ドリフト (期待成長率)  $\mu$  は以下の一次関数と設定する。

$$\mu(y) = a + by \quad (3)$$

但し、 $a, b$  を定数とする。

### (3) 安全資産モデルの設定

$S^0$  は安全資産とし、次の常微分方程式に従うとする。

$$\frac{dS_t^0}{S_t^0} = r(Y_t)dt \quad (4)$$

但し、金利  $r$  を以下の一次関数と設定する。

$$r(y) = a_0 + b_0y. \quad (5)$$

但し、 $a_0, b_0$  を定数とする。

### (4) 株と安全資産のポートフォリオ

$X_t$  が時刻  $t$  での投資額、 $\pi_t$  が株への投資割合、 $(1 - \pi_t)$  が安全資産への投資割合とし、自己資金調達条件 (*self-financing condition*) によると、ポートフォリオが従う確率微分方程式は以下ようになる。

$$\begin{aligned} dX_t^\pi &= X_t^\pi \left[ \pi_t \frac{dS_t}{S_t} + (1 - \pi_t) \frac{dS_t^0}{S_t^0} \right] \\ &= X_t^\pi [(r(Y_t) + \pi_t \bar{\mu}(Y_t))dt + \pi_t \sigma dW_t]. \end{aligned} \quad (6)$$

$$\text{但し, } \bar{\mu}(y) = \mu(y) = r(y) = \bar{a} + \bar{b}y \quad (7)$$

$$\bar{a} = a - a_0, \bar{b} = b - b_0 \quad (8)$$

とする。

(5) 基準値 (benchmark) のモデル設定

$I$  を基準値とし、次の確率微分方程式に従うとする;

$$\frac{dI_t}{I_t} = \eta(Y_t)dt + r_0dB_t + r_1dW_t + r_2dM_t \quad (9)$$

$r_0, r_1, r_2$  が定数で、 $M$  は  $B$  と  $W$  に独立な標準ブラウン運動である。 $\eta$  は以下の 1 次関数と設定する。

$$\eta(y) = a_I + b_I y, \quad (10)$$

但し、 $a_I$  と  $b_I$  は定数とする。シミュレーションの時は、 $I=1$  とする。

(6) ポートフォリオと基準値の比率

ここではポートフォリオと基準値の比率  $Z$  と考える;

$$Z_t^\pi = \frac{X_t^\pi}{I_t}. \quad (11)$$

シミュレーション時、 $I$  を 1 にするから、

$$Z_t^\pi = X_t^\pi \quad (12)$$

となる。伊藤の公式によると  $\ln Z_T^\pi$  が

$$\begin{aligned} \ln Z_T^\pi = & \int_0^T (r(Y_t) - \eta(Y_t) + \frac{1}{2}r^2 + \pi_t \bar{\mu}(Y_t) \\ & - \frac{1}{2}(\pi_t \sigma)^2)dt + \int_0^T (\pi_t \sigma - r_1)dW_t - r_0B_T - r_2M_T \end{aligned} \quad (13)$$

となる。

(7) 投資額の成長率

投資額の成長率  $L$  を次のようにおく;

$$L_t^\pi = \frac{\ln Z_t^\pi}{t} \quad (14)$$

式 (14) は  $Z_t^\pi = e^{L_t^\pi t}$  と書け、 $L$  が成長率を表すことがわかる。

(8) アップサイドチャンスの問題設定

目標成長率  $c$  を設定し、成長率  $L$  が  $c$  を上回る確率を最大にする問題が、

$$\sup_{\pi \in A} \limsup_{T \rightarrow \infty} \frac{1}{T} \ln P[L_T^\pi \geq c] \quad (15)$$

となる。 $A$  は実行可能なポートフォリオの集合とする。 $T \rightarrow \infty$  であるため、長時間問題を考えている。 $T$  を十分大きいとすると、式 (15) は形式的に、

$$P[L_T^\pi \geq c] \approx \exp(-\Gamma(c, \pi)T) \quad (16)$$

と書ける。但し、 $\Gamma(c, \pi)$  がポートフォリオ  $\pi$  のレート関数である。アップサイドチャンスポートフォリオ最適化とは、このレート関数  $\Gamma(c, \pi)$  を最小にするポートフォリオを考えることである。

3. アップサイドチャンスの問題設定

目標成長率  $c$  を設定し、成長率  $L$  が  $c$  を上回る確率を最大にする問題が、

$$P[L_T^\pi \geq c] \approx \exp(-\Gamma^*(c, \pi)T) \quad (17)$$

となる。 $\Gamma^*(c, \pi)$  がポートフォリオ  $\pi$  のレート関数である。 $\pi$  が  $A$  に属する。 $(A : \text{the set of admissible portfolio control})$

大偏差原理によると、アップサイトチャンスの確率を無限大まで持っていくと、

$$\limsup_{T \rightarrow \infty} \frac{1}{T} \ln P[L_T^\pi \geq c] \quad (18)$$

となる。これは、アップサイトチャンスの問題設定となる。

### (1)Pham[1] の主結果

式 (18) がレート関数によって変換される。次の式となる。

$$\Lambda(\lambda, \pi) = \limsup_{T \rightarrow \infty} \frac{1}{T} \ln E[\exp(\lambda T L_T^\pi)] \quad (19)$$

式 (19) についての証明を行う。Dembo and Zeitouni 1998 の定理 2.3.6 より、

$$\Lambda_n(\lambda) \doteq \log E[\exp(\lambda, Z_n)]$$

$$\frac{1}{n} \Lambda_n(\lambda_n) = \frac{1}{n} \log E[\exp(\lambda_n, Z_n)]$$

$\forall \lambda \in R$  の時、 $\Lambda(\lambda)$  の定義が次のようにとなる。

$$\Lambda(\lambda) = \limsup_{n \rightarrow \infty} \frac{1}{n} \Lambda_n(n\lambda)$$

$\Lambda(\lambda)$  の定義より、

$$\Lambda(\lambda) = \limsup_{n \rightarrow \infty} \frac{1}{n} \log E[\exp(n\lambda Z_n)]$$

本論文の符号とそろえると、式 (19) が証明できた。

$\pi$  を固定し、独立変数  $\lambda$  が双対性を持っている。式 (19) が以下の式となる。

$$\limsup_{T \rightarrow \infty} \frac{1}{T} \ln P[L_T^\pi \geq c] = - \sup_{\lambda \geq 0} [\lambda c - \Lambda(\lambda, \pi)] \quad (20)$$

大偏差原理による式 (18) の確率を最大化するため、式 (18) の確率のレート関数が次のような関係性を作る。

$$\Lambda(\lambda) = \sup_{\pi \in A} \Lambda(\lambda, \pi) \quad (21)$$

ここで、 $\lambda \geq 0$  を固定すると、式 (21) が ergodic risk-sensitive control problem となる。大偏差原理による主結果が式 (21) と双対性を持っている。次の式となる。

$$J(c) = \sup_{\pi \in A} \limsup_{T \rightarrow \infty} \frac{1}{T} \ln P[L_T^\pi \geq c], c \in R. \quad (22)$$

定理 1.1 (Pham[1] の Theorem 3.1)  $\forall \lambda \in [0, \bar{\lambda}], \bar{\lambda} \in (0, \infty)$  が存在すると仮定すると、式 (19) と式 (21) が双対性を持っている。戦略  $\pi(\hat{\lambda})$  が次の式となる。

$$\Lambda(\lambda) = \limsup_{T \rightarrow \infty} \frac{1}{T} \ln E[\exp(\lambda T L_T^{\hat{\pi}(\lambda)})] \quad (23)$$

$\Lambda(\lambda)$  が連続な差分だと仮定するとき、 $J(c)$  が

$$J(c) = - \sup_{\lambda \in [0, \bar{\lambda}]} [\lambda c - \Lambda(\lambda)] \quad (24)$$

となる。さらに、戦略の解が

$$\pi_t^{*,n} = \begin{cases} \hat{\pi}_t(\lambda(c + \frac{1}{n})), & c > \Lambda'(0) \\ \hat{\pi}_t(\lambda(\Lambda'(0) + \frac{1}{n})), & c \leq \Lambda'(0) \end{cases} \quad (25)$$

となる。最適戦略解が

$$\limsup_{n \rightarrow \infty} \limsup_{T \rightarrow \infty} \frac{1}{T} \ln P[L_T^{\pi^{*,n}} \geq c] = J(c), c \in R. \quad (26)$$

となる。

(2)Pham[1] の主結果についての証明

凸関数  $\Lambda \in [0, \bar{\lambda})$  の Fenchel-Legendre transform が

$$\Lambda^*(c) = \sup_{\lambda \in [0, \bar{\lambda})} [\lambda(c) - \Lambda(\lambda)], c \in R. \quad (27)$$

となる。Lemma2.3.9 in Dembo and Zeitouni 1998 より、関数  $\Lambda^*$  が狭義凸関数で増加非減少である。以下の式を満たす。

$$\Lambda^*(c) = \begin{cases} \lambda(c)c - \Lambda(\lambda(c)), & c > \Lambda'(0), \\ 0, & c \leq \Lambda'(0), \end{cases} \quad (28)$$

$$\lambda(c)c - \Lambda^*(c) > \lambda(d)c - \Lambda^*(d), \quad (29)$$

$\forall \Lambda'(0) < c, \forall d \neq c$ , ここで、 $\lambda(c) \in (0, \bar{\lambda}), \Lambda'(\lambda(c)) = c \in (\Lambda'(0), \infty)$  である。さらに、 $\Lambda^*$  が  $(-\infty, \Lambda'(\bar{\lambda})) = R$  において連続的である。

## 2.1 上限 (Upper bound)

$\forall c \in R, \pi \in A$ , Chebycheff の不等式を応用すると、

$$P[L_T^\pi] \leq \exp(-\lambda c T) E[\exp(\lambda T L_T^\pi)] \quad (30)$$

となり、無限大を付けると、

$$\limsup_{T \rightarrow \infty} \frac{1}{T} \ln P[L_T^\pi] \leq -\lambda c + \limsup_{T \rightarrow \infty} \frac{1}{T} E[\exp(\lambda T L_T^\pi)] \quad (31)$$

となる。 $\Lambda$  と  $\Lambda^*$  の定義によると、

$$\sup_{\pi \in A} \limsup_{T \rightarrow \infty} \frac{1}{T} \ln P[L_T^\pi \geq c] \leq -\Lambda^*(c) \quad (32)$$

となる。

## 2.2 下限 (Lower bound)

$c \in R, (\Omega, F_t)$  上で確率測度  $Q_T^n$  を定義すると、

$$\frac{dQ_T^n}{dP} = \exp[\lambda(c_n) T L_T^{\pi^{*,n}} - \Lambda_T(\lambda(c_n), \pi^{*,n})], \quad (33)$$

ここで、

$$c_n = \begin{cases} c + \frac{1}{n}, & c > \Lambda'(0), \\ \Lambda'(0) + \frac{1}{n}, & \text{その他}, \end{cases}$$

そうして、

$$\Lambda_T(\lambda, \pi) = \ln E[\exp(\lambda T L_T^\pi)], \quad (34)$$

$\lambda \in [0, \bar{\lambda}), \pi \in A$ .  $\epsilon$ をできる限り小さくすると、 $c \leq c_n - \epsilon$ となる。そうすると、

$$\begin{aligned} \frac{1}{T} \ln P \left[ L_T^{\pi^*, n} \geq c \right] &\geq \frac{1}{T} \ln P \left[ c_n - \epsilon < L_T^{\pi^*, n} < C_n + \epsilon \right] \\ &= \frac{1}{T} \ln \left( \int \frac{dP}{dQ_T^n} 1_{c_n - \epsilon < L_T^{\pi^*, n} < c_n + \epsilon} dQ_T^n \right) \\ &\geq -\lambda(c_n)(c_n + \epsilon) + \frac{1}{T} \Lambda_T(\lambda(c_n), \pi^*, n) + \frac{1}{T} \ln Q_T^n \left[ c_n - \epsilon < L_T^{\pi^*, n} < c_n + \epsilon \right] \end{aligned}$$

式 (23) より

$$\begin{aligned} \liminf_{T \rightarrow \infty} \frac{1}{T} \ln P \left[ L_T^{\pi^*, n} \geq c \right] &\geq -\lambda(C_n)(c_n + \epsilon) + \Lambda(\lambda(c_n)) + \liminf_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ c_n - \epsilon < L_T^{\pi^*, n} < c_n + \epsilon \right] \\ &\geq -\Lambda^*(c_n) - \lambda(c_n)\epsilon + \liminf_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ c_n - \epsilon < L_T^{\pi^*, n} < c_n + \epsilon \right] \end{aligned} \quad (35)$$

式 (35) の第 2 の不等式が  $\Lambda^*$  の定義と式 (28) より、

$$\liminf_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ c_n - \epsilon < L_T^{\pi^*, n} < c_n + \epsilon \right] = 0$$

となる。確率測度  $Q_T^n$  の下で、 $\tilde{\Lambda}_T^n$  が式 (33) より以下の式となる。

$$\begin{aligned} \tilde{\Lambda}_T^n(\zeta) &:= \ln E^{Q_T^n} \left[ \exp \left( \zeta T L_T^{\pi^*, n} \right) \right] \\ &= \Lambda_T(\lambda(c_n) + \zeta, \pi^*, n) - \Lambda_T(\lambda(c_n), \pi^*, n) \end{aligned}$$

さらに、式 (21) と式 (23) の定義より、 $\forall \zeta \in [-\lambda(c_n), \bar{\lambda} - \lambda(c_n))$  :

$$\limsup_{T \rightarrow \infty} \frac{1}{T} \tilde{\Lambda}_T^n(\zeta) \leq \Lambda(\lambda(c_n) + \zeta) - \Lambda(\lambda(c_n)) \quad (36)$$

上限で使われた Chebycheff の不等式より、 $\forall \zeta \in [-\lambda(c_n), \bar{\lambda} - \lambda(c_n))$  の場合は

$$\begin{aligned} \limsup_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ L_T^{\pi^*, n} \geq c_n + \epsilon \right] &\leq -\zeta(c_n + \epsilon) + \limsup_{T \rightarrow \infty} \frac{1}{T} \tilde{\Lambda}_T^n(\zeta) \\ &\leq -\zeta(c_n + \epsilon) + \Lambda(\zeta + \lambda(C_n)) - \Lambda(\lambda(c_n)) \end{aligned}$$

上式を推定すると、以下の式となる。

$$\begin{aligned} \limsup_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ L_T^{\pi^*, n} \geq C_n + \epsilon \right] &\leq -\sup\{\zeta(c_n + \epsilon) - \Lambda : \zeta \in [\lambda(c_n), \bar{\lambda}]\} - \Lambda(\lambda(c_n)) + \lambda(c_n)(c_n + \epsilon) \\ &\leq -\Lambda^*(c_n + \epsilon) - \Lambda(\lambda(c_n)) + \lambda(c_n)(c_n + \epsilon) \\ &= -\Lambda^*(c_n + \epsilon) + \Lambda^*(c_n) + \epsilon\lambda(c_n) \end{aligned} \quad (37)$$

ここで、式 (28) により、二番目の不等式が成り立つ。同様に、 $\forall \zeta \in [-\lambda(c_n), 0)$  の場合は

$$\begin{aligned} \limsup_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ L_T^{\pi^*, n} \leq c_n - \epsilon \right] &\leq -\zeta(c_n - \epsilon) + \limsup_{T \rightarrow \infty} \frac{1}{T} \tilde{\Lambda}_T^n(\zeta) \\ &\leq -\zeta(c_n - \epsilon) + \Lambda(\lambda(c_n) + \zeta) - \Lambda(\lambda(c_n)) \end{aligned}$$

さらに、

$$\begin{aligned} \limsup_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ L_T^{\pi^*, n} \leq c_n - \epsilon \right] &\leq -\sup\{\zeta(c_n - \epsilon) - \Lambda : \zeta \in [\lambda(c_n), \bar{\lambda}]\} \\ &\quad - \Lambda(\lambda(c_n)) + \lambda(c_n)(c_n + \epsilon) - \Lambda(\lambda(c_n)) + \lambda(c_n)(c_n - \epsilon) \\ &\leq -\Lambda^*(c_n - \epsilon) + \Lambda^*(c_n) - \epsilon\lambda(c_n) \end{aligned} \quad (38)$$

式 (37) と式 (38) より、

$$\begin{aligned}
& \limsup_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ \{L_T^{\pi^{*,n}} \leq c_n - \epsilon\} \cup \{L_T^{\pi^{*,n}} \geq c_n + \epsilon\} \right] \\
& \leq \max \left\{ \limsup_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ L_T^{\pi^{*,n}} \geq c_n + \epsilon \right]; \limsup_{T \rightarrow \infty} \frac{1}{T} \ln Q_T^n \left[ L_T^{\pi^{*,n}} \leq c_n - \epsilon \right] \right\} \\
& \leq \max \{ -\Lambda^*(c_n + \epsilon) + \Lambda^*(c_n) + \epsilon \lambda(c_n); -\Lambda^*(c_n - \epsilon) + \Lambda^*(c_n) - \epsilon \lambda(c_n) \} < 0
\end{aligned}$$

を推定することができる。

$$Q_T^n \left[ \{L_T^{\pi^{*,n}} \leq c_n - \epsilon\} \cup \{L_T^{\pi^{*,n}} \geq c_n + \epsilon\} \right] \rightarrow 0$$

にすると、従って、

$$T \rightarrow \infty \text{ のとき, } Q_T^n [c_n - \epsilon < L_T^{\pi^{*,n}} < c_n + \epsilon] \rightarrow 1 \quad (39)$$

となる。式 (35) の  $\epsilon \rightarrow 0$  すると、以下の式が得られる。

$$\liminf_{T \rightarrow \infty} \frac{1}{T} \ln P[L_T^{\pi^{*,n}} \geq c] \geq -\Lambda^*(c_n). \quad (40)$$

式 (32) より、

$$\liminf_{n \rightarrow \infty} \liminf_{T \rightarrow \infty} \frac{1}{T} \ln P[L_T^{\pi^{*,n}} \geq c] \geq -\Lambda^*(c). \quad (41)$$

従って、証明が終了となる。

#### 4. ファクターモデルの場合分け

H.Pham の結果に基づき、数値的シミュレーションを行う。数値実験に関して、ファクターモデルに2つの場合を考える。

##### (1) ファクターモデルが定数

ファクターが定数の時で、株価がブラックショールズと同じになる。そのとき、株価が

$$\frac{dS_t}{S_t} = a dt + \sigma dW_t \quad (42)$$

となる。 $a, \sigma$  が定数である。

安全資産が

$$\frac{dS_t^0}{S_t^0} = a_0 dt \quad (43)$$

となる。 $a_0$  が定数である。

##### 1.1 Black-Scholes model の戦略

株価がブラックショールズの時、 $I \equiv 1, b_0 = b = 0$ , そうすると、 $\bar{c} = 0$ .

大偏差原理による解の関数は、

$$J(c) = \begin{cases} -(\sqrt{c} - \sqrt{\bar{c}})^2, & \text{if } c \geq \bar{c} \\ 0, & \text{if } c < \bar{c}. \end{cases} \quad (44)$$

となる。ここで、 $\bar{c} = \frac{1}{2}a^2$ .  $c < \bar{c}$  の時、ポートフォリオの最適戦略が次のように与えられる。

$$\pi_t^* = \bar{a}, \quad (45)$$

$c \geq \bar{c}$  の時、ポートフォリオの最適戦略が次のように与えられる。

$$\pi_t^* = \sqrt{2c}. \quad (46)$$

##### (2) ファクターモデルが Vasicek interest rate model

Vasicek interest rate model の時、

$$r(y) = a_0 + b_0 y, b_0 \neq 0.$$

$$\mu(y) = a + by, b \neq 0.$$

##### 2.1 Vasicek interest rate model の戦略

ベンチマーク I が 1 とする。 $c < \bar{c}$  の時、ポートフォリオの最適戦略が次のように与えられる。

$$\pi_t^* = \bar{b}Y_t + \bar{a}, \quad (47)$$

$c \geq \bar{c}$  の時、ポートフォリオの最適戦略が次のように与えられる。

$$\pi_t^* = \frac{\bar{b}Y_t}{1 - \lambda c} + \bar{a}. \quad (48)$$

ここで、 $\bar{c} = \frac{a^2}{2} + \frac{b^2}{4k}$  とする。

## 5. 数値実験の方法

ブラウン運動の入っている連続モデルが離散化しなければ、数値実験することができない。離散化するため、オイラ-丸山近似方法を用いた。

(1) Black-Sholes model のオイラ-丸山近似

### 1.1 株価の確率微分方程式のオイラ-丸山近似

$0 = t_0 < t_1 < \dots < tm$  として、ブラックショールズモデルの確率微分方程式が

$$\frac{dS_t}{S_t} = a dt + \sigma dW_t \quad (49)$$

次のようになる。

$$dS(t) = S(t + \Delta t) - S(t) \quad (50)$$

$$dW(t) = W(t + \Delta t) - W(t) \quad (51)$$

より、

$$S(t + \Delta t) - S(t) = aS(t)\Delta t + \sigma S(t)(W(t + \Delta t) - W(t))$$

ブラウン運動の性質より、

$$W(t + \Delta t) - W(t) = \sqrt{\Delta t}Z \quad (52)$$

となる。ここで、 $Z_1, Z_2, \dots$  は互いに独立な標準正規分布に従う確率変数である。

よって、ブラックショールズモデルの確率微分方程式が

$$S(t + \Delta t) = aS(t)\Delta t + \sigma S(t)\sqrt{\Delta t}Z \quad (53)$$

となる。

### 1.2 ポートフォリオの確率微分方程式のオイラ-丸山近似

同上に、ポートフォリオの確率微分方程式をオイラ-丸山近似すると、

$$dX_t = X_t[(r(Y_t) + \pi_t \bar{\mu}(Y_t))dt + \pi_t \sigma dW_t] \quad (54)$$

をオイラ-丸山近似すると、が次の式となる。

$$X(t + \Delta t) = X(t) + X(t)[(r(Y_t) + \pi_t \bar{\mu}(Y_t))dt + \pi_t \sigma \sqrt{\Delta t}Z] \quad (55)$$

(2) Vasicek model のオイラ-丸山近似

### 2.1 ファクターモデルの確率微分方程式のオイラ-丸山近似

ファクターモデルに Vasicek model が入っているため、ファクターモデルの丸山近似もする必要がある。ファクターモデルの確率微分方程式が

$$dY_t = -k(Y_t - \bar{y})dt + v dB_t \quad (56)$$

を丸山近似すると、ファクターモデルの確率微分方程式のオイラ-丸山近似が

$$Y(t + \Delta t) = Y(t) - k(Y_t - \bar{y})dt + v\sqrt{\Delta t}Z \quad (57)$$

となる。

## 2.2 株価の確率微分方程式のオイラ-丸山近似

Vasicek model の株価にファクターが入っているため、上式のファクター確率微分方程式を代入しなければなりません。

$$S(t + \Delta t) - S(t) = (Y(t) - k(Y_t - \bar{y}))dt + v\sqrt{\Delta t}Z_1 S(t)\Delta t + \sigma S(t)\sqrt{\Delta t}Z_2$$

ここで、 $Z_1$  と  $Z_2$  は互いに独立な標準正規分布に従う確率変数である。

## 2.3 ポートフォリオの確率微分方程式のオイラ-丸山近似

同じように、ポートフォリオのファクターモデルが Vasicek model が入っているため、ファクター確率微分方程式を代入しなければなりません。ポートフォリオの確率微分方程式のオイラ-丸山近似が次のようになる。

$$X(t + \Delta t) = X(t) + X(t)[(r(Y(t) - k(Y_t - \bar{y}))dt + v\sqrt{\Delta t}Z_1) + \pi_t \bar{\mu}(Y(t) - k(Y_t - \bar{y}))dt + v\sqrt{\Delta t}Z_1)]dt + \pi_t \sigma \sqrt{\Delta t}Z_3]$$

ここで、 $Z_1$  と  $Z_3$  は互いに独立な標準正規分布に従う確率変数である。

上記、オイラ丸山近似をモンテカルロ法で実験する。具体的に、100 万通りのシナリオを作り、満期時に、成長率が目標成長率  $c$  を超えたシナリオをカウントし、100 万通りに占める割合を計算する。

### (3) 数値実験のモデル係数について

数値実験では、モデルの係数が自分で設定する。設定の基準は、実際の市場を参考にしている。例えば、企業の成長率  $a$  に関しては、成長の著しい企業もある一方、負の収益を持っている企業も存在する。表でまとめると、以下のようになる。

表 1 企業の成長率

企業の成長率 a	企業の状況を表す
-0.1	負の収益がある企業
0	成長が止まっている企業
0.1	緩やかに成長する企業
0.5	著しく成長する企業

表 2 企業のボラティリティ

企業のボラティリティ	株価の変動を表す
0.2	株価の変動幅が小さい
0.7	株価の変動幅が大きい

注意 3.1 数値実験では、 $c \geq \bar{c}$  の場合のみについて行った。 $c < \bar{c}$  の場合は  $c$  を上回る確率が 0 となるからである。

## 6. 数値実験の結果と考察

### (1) 株価が Black-Scholes Model の数値実験の結果

$a = -0.1, 0, 0.1, 0.5; \sigma = 0.2, 0.5$  とする。

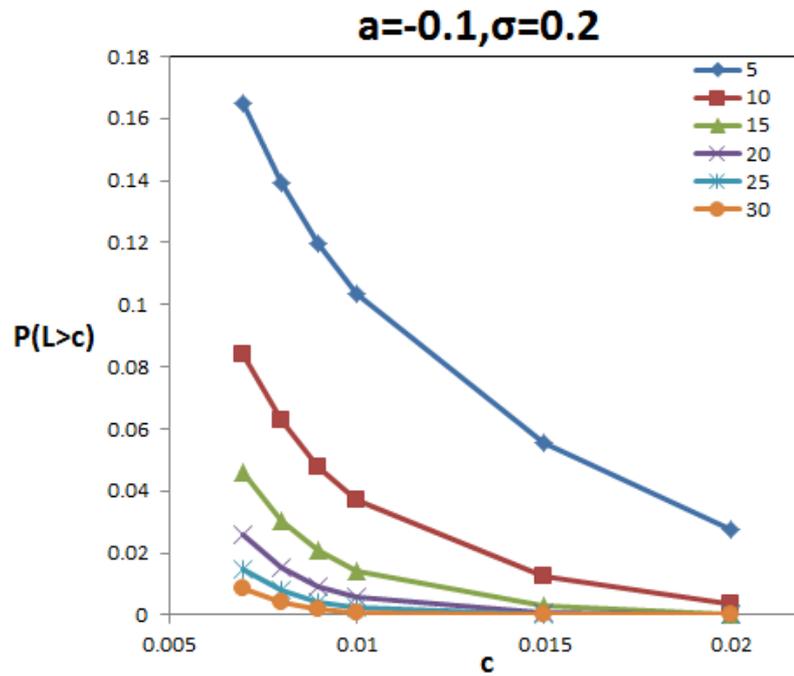


図 1 株価が Black-Scholes Model である ( $a=-0.1, \sigma=0.2$ )

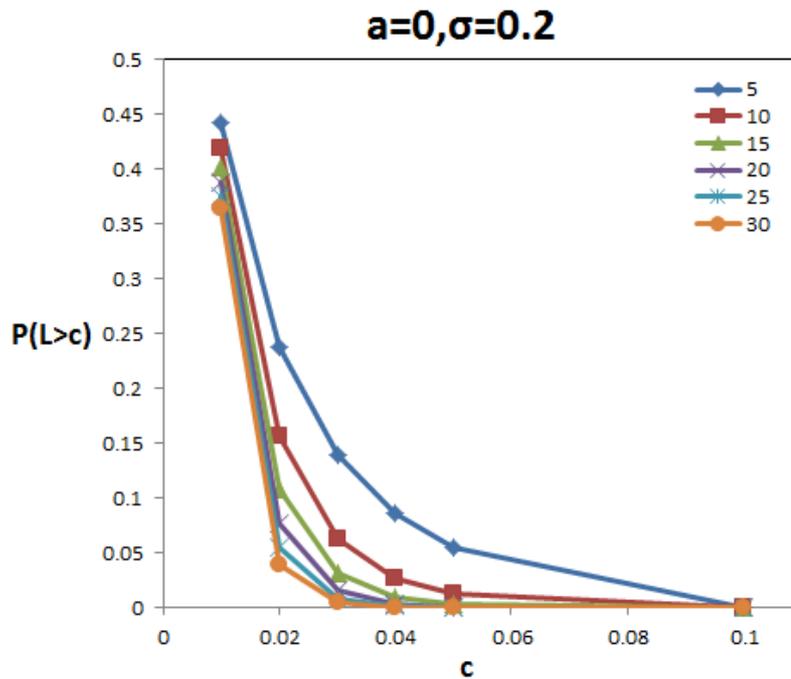


表 3 株価が Black-Scholes Model である ( $a=0, \sigma=0.2$ )

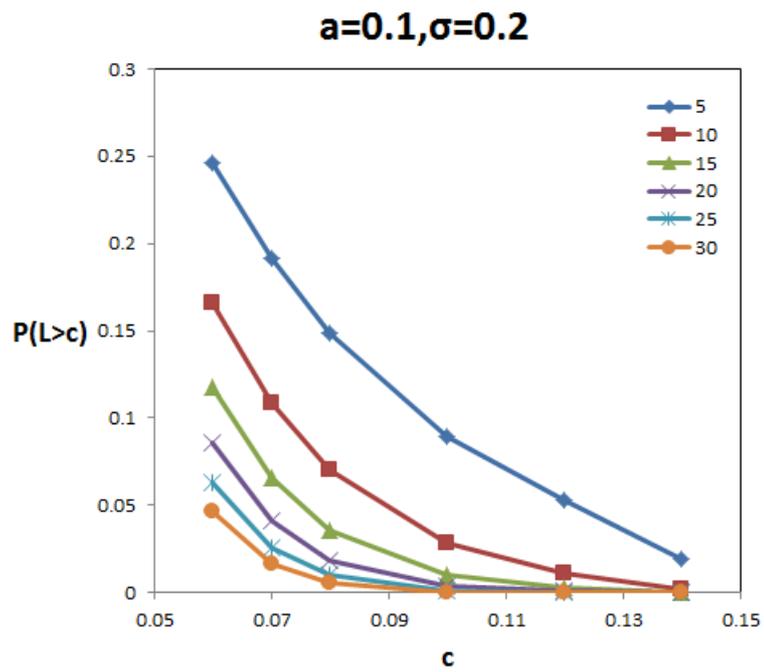


表 4 株価が Black-Scholes Model である (a=0.1,σ=0.2)

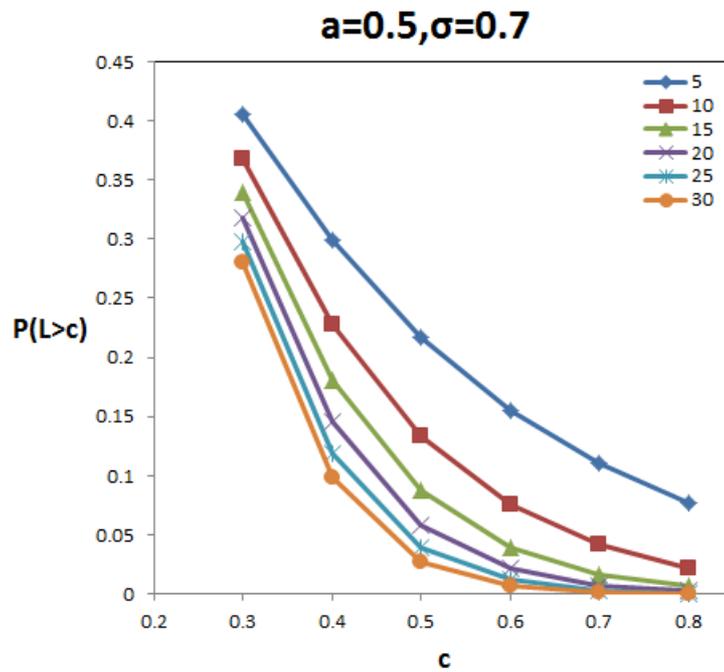


表 5 株価が Black-Scholes Model である (a=0.5,σ=0.7)

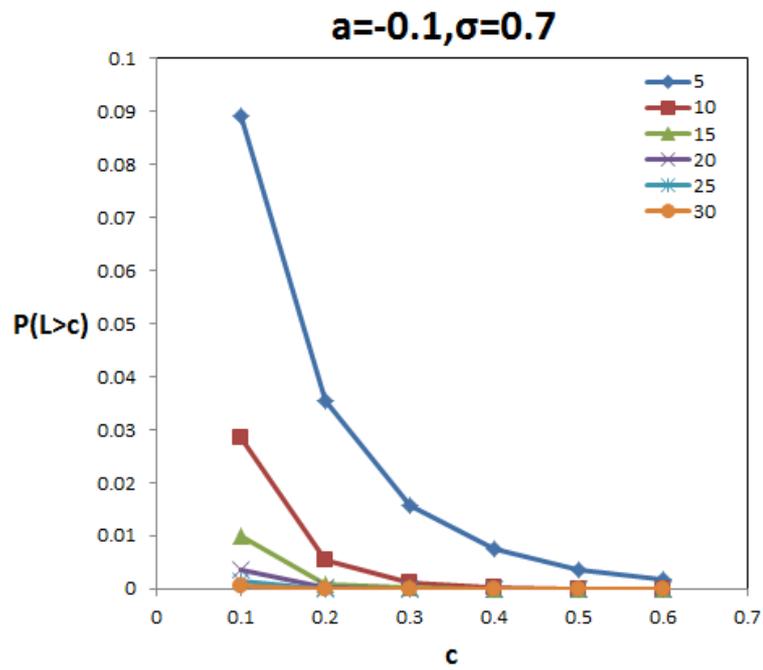


表 6 株価が Black-Scholes Model である ( $a=-0.1, \sigma=0.7$ )

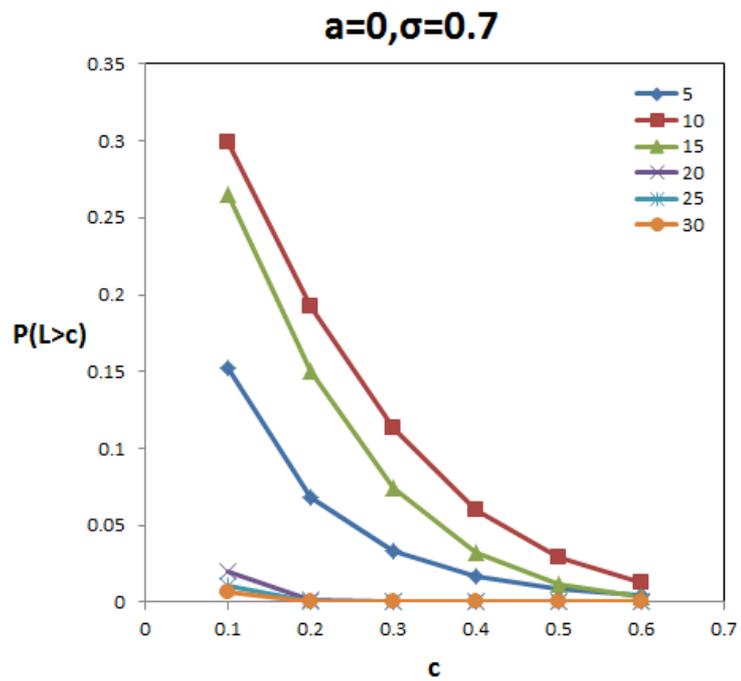


表 7 株価が Black-Scholes Model である ( $a=0, \sigma=0.7$ )

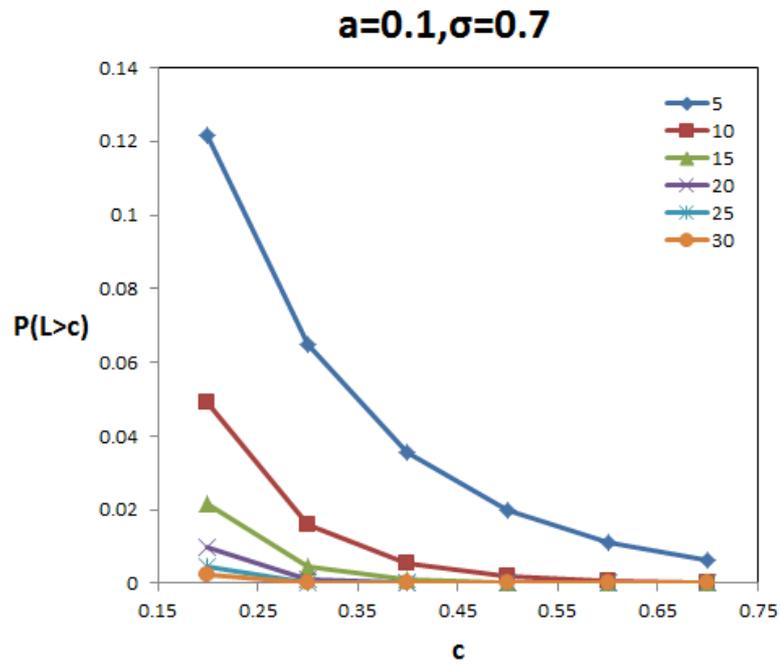


表 8 株価が Black-Scholes Model である (a=0.1, σ=0.7)

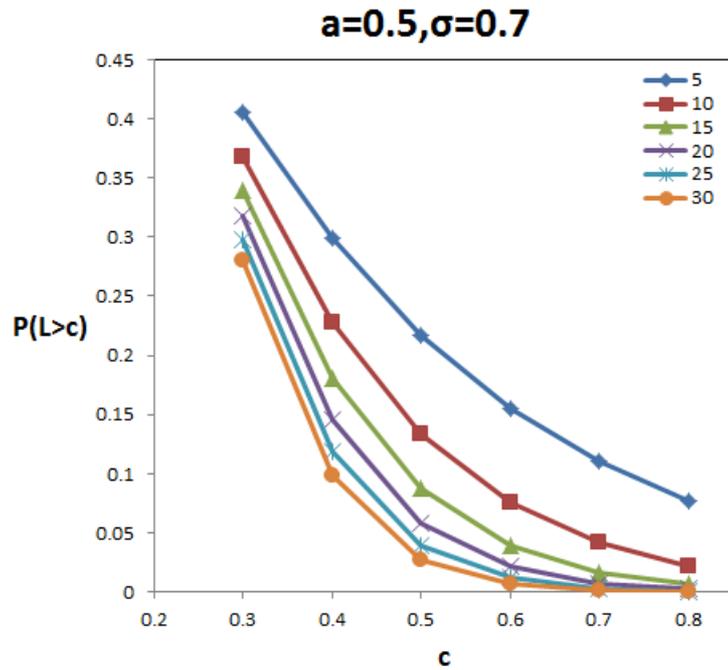


表 9 株価が Black-Scholes Model である (a=0.5, σ=0.7)

(2) 株価が Black-Scholes Model の数値実験の結果についての考察

株価が Black-Scholes Model の時、戦略  $\pi$  に変数が入っていない。比較的簡単なモデルになる。 $P[L_T^{\pi} \geq c] \approx \exp(-\Gamma^*(c, \pi)T)$  となっているから、時間  $T$  が大きくなるほど、確率が下がっていく傾向である。従って、グラフの形も予測通りとなっている。

(3) Black-Scholes Model の理論値と実験値の誤差

数値実験の精度を確かめる。ブラック・ショールズモデルの場合、理論値を具体的に計算することができた。その理論値とシミュレーションで得た数値を比較し、誤差がどのくらい生じるかを把握する。以下がブラックショールズの理論値である。

表 10 理論値と実験値とその誤差 ( $a=-0.1, \sigma = 0.2$ )

理論確率	T/c	0.00606	0.007	0.008	0.009	0.01	0.015	0.02
	5	0.196462	0.165321	0.139772	0.119697	0.103572	0.055769	0.033359
	10	0.113485	0.084445	0.063101	0.048073	0.037214	0.01221	0.004756
	15	0.069472	0.045997	0.030538	0.020787	0.014448	0.002924	0.000747
	20	0.043758	0.025849	0.015283	0.009311	0.005819	0.00073	0.000123
	25	0.028045	0.014804	0.007804	0.00426	0.002396	0.000187	2.07E-05
	30	0.01819	8.59E-03	0.00404	0.001977	0.001001	4.85E-05	3.54E-06

実験値	T/c	0.00606	0.007	0.008	0.009	0.01	0.015	0.02
	5	0.196042	0.16491	0.139444	0.119615	0.103447	0.055592	0.027742
	10	0.113258	0.084158	0.062814	0.047762	0.036976	0.012247	0.003373
	15	0.069481	0.046252	0.030602	0.02077	0.014419	0.002897	0.000493
	20	0.043817	0.025865	0.015203	0.009283	0.005814	0.000774	0.000074
	25	0.028084	0.014677	0.007815	0.004189	0.00237	0.000184	0.000009
	30	0.018088	0.00857	0.004092	0.002053	0.001038	0.000032	0.000002

誤差	T/c	0.00606	0.007	0.008	0.009	0.01	0.015	0.02
	5	1.002144	1.002492	1.002351	1.000684	1.001204	1.003189	1.202455
	10	1.002008	1.003414	1.004567	1.00652	1.006435	0.996968	1.410081
	15	0.99987	0.994477	0.99791	1.000805	1.001984	1.009149	1.515214
	20	0.998649	0.999374	1.005231	1.003029	1.000901	0.942785	1.657214
	25	0.998624	1.008653	0.998577	1.01693	1.010916	1.014339	2.295279
	30	1.005621	1.002106	0.98719	0.962901	9.64E-01	1.516027	1.769537

誤差式=理論値/実験値とする。計算された誤差は以下の通りである。理論値と実験値の比が1に近ければ近いほど、誤差が小さいとなる。下の表からみると、 $a = 0.1, \sigma = 0.2$  の時、 $c$  が0.06 から0.12まで誤差はほぼ1である。理論値と実験値の差がほぼないと考えられる。数値実験の精度が高いと判断できる。但し、 $c$  が0.14の時、誤差が大きくなる。 $a = 0.5, \sigma = 0.2$  の時、誤差は  $a = 0.1, \sigma = 0.2$  の時と同様に、 $c$  が小さいとき、誤差が小さく、 $c$  が大きいとき、誤差が大きくなる。このような現象が起きる理由はシミュレーションの手法に問題があると考えられる。数値実験で用いる方法が丸山近似法とモンテカルロ法である。 $c$  が小さいとき、 $c$  を超える確率が大きく、多少の誤差があっても、確率に影響が出ない。但し、 $c$  が大きいとき、 $c$  を上回る確率がかなり小さくなった。その時誤差が生ずると、値のぶれが大きくなる。従って、理論値と実験値の比が大きくなる。

表 11 理論値と実験値とその誤差 (a=0,σ=0.2)

理論確率	T/c	0.01	0.02	0.03	0.04	0.05	0.1
	5	0.442977	0.237137	0.140194	0.086772	0.055163	0.006912
	10	0.419632	0.155786	0.063434	0.027139	0.011969	0.000249
	15	0.401904	0.107607	0.030766	0.009208	0.002841	1E-05
	20	0.387113	0.076203	0.015429	0.003246	0.000703	4.25E-07
	25	0.374215	0.054799	0.007895	0.001171	0.000178	1.85E-08
	30	0.362674	0.039826	0.004096	0.000429	4.59E-05	8.19E-10

実験値	T/c	0.01	0.02	0.03	0.04	0.05	0.1
	5	0.442433	0.236854	0.139866	0.086559	0.054964	0.000378
	10	0.419324	0.155713	0.063149	0.026946	0.012006	0.000001
	15	0.402085	0.107204	0.030846	0.009228	0.002823	0
	20	0.387771	0.076339	0.015372	0.003295	0.000734	0
	25	0.374657	0.054877	0.007895	0.001192	0.000173	0
	30	0.363328	0.039866	0.004145	0.000442	0.000003	0

誤差	T/c	0.01	0.02	0.03	0.04	0.05	0.1
	5	1.001229	1.001196	1.002347	1.002463	1.003625	18.28577
	10	1.000734	1.000469	1.004517	1.007171	0.996942	249.2922
	15	0.99955	1.00376	0.997399	0.997813	1.006303	-
	20	0.998302	0.99822	1.003723	0.985223	0.957637	-
	25	0.99882	0.998584	1.000049	0.982284	1.030197	-
	30	0.998199	0.998989	0.988059	0.97008	1.530848	-

表 12 理論値と実験値とその誤差 (a=0.1,σ =0.2)

理論値	T/c	0.06	0.07	0.08	0.1	0.12	0.14
	5年	0.2467	0.1921	0.1492	0.0895	0.0533	0.0315
	10年	0.1663	0.1092	0.0707	0.0287	0.0112	0.0043
	15年	0.1177	0.0659	0.0359	0.0100	0.0026	0.0006
	20年	0.0854	0.0409	0.0188	0.0036	0.0006	0.0001
	25年	0.0628	0.0258	0.0100	0.0013	0.0002	0.0000
	30年	0.0467	0.0165	0.0054	0.0005	0.0000	0.0000

実験値	T/c	0.06	0.07	0.08	0.1	0.12	0.14
	5年	0.2464	0.1915	0.1489	0.0892	0.0531	0.0192
	10年	0.1662	0.1089	0.0705	0.0285	0.0113	0.0017
	15年	0.1172	0.0659	0.0359	0.0099	0.0026	0.0002
	20年	0.0854	0.0410	0.0188	0.0037	0.0007	0.0000
	25年	0.0628	0.0257	0.0100	0.0013	0.0002	0.0000
	30年	0.0467	0.0164	0.0055	0.0005	0.0000	0.0000

誤差	T/c	0.06	0.07	0.08	0.1	0.12	0.14
	5年	1.0012	1.0029	1.0024	1.0031	1.0031	1.6418
	10年	1.0012	1.0026	1.0038	1.0080	0.9983	2.5247
	15年	1.0042	0.9991	0.9994	1.0044	1.0092	3.7316
	20年	0.9991	0.9978	1.0013	0.9852	0.9550	5.0264
	25年	0.9997	1.0033	1.0049	0.9940	1.0004	16.1582
	30年	1.0003	1.0094	0.9823	0.9984	1.4306	2.6416

表 13 理論値と実験値とその誤差 ( $a=0.5, \sigma=0.2$ )

理論確率	T/c	0.4	0.5	0.6	0.7	0.8	0.9
	5	0.656661	0.411532	0.215277	0.09576	0.036956	0.012601
	10	0.715813	0.375915	0.13248	0.032366	0.005745	0.000774
	15	0.757615	0.349268	0.086083	0.011842	0.000983	5.29E-05
	20	0.79009	0.32736	0.057457	0.004498	0.000176	3.79E-06
	25	0.81646	0.308538	0.038987	0.001747	3.22E-05	2.79E-07
	30	0.838434	0.291941	0.026754	0.000689	6.00E-06	2.09E-08

実験値	T/c	0.4	0.5	0.6	0.7	0.8	0.9
	5	0.65557	0.409025	0.213048	0.094259	0.036071	0.01228
	10	0.714145	0.372919	0.13021	0.03133	0.005529	0.000727
	15	0.755774	0.345724	0.08399	0.011377	0.000929	0.000054
	20	0.787781	0.324199	0.056039	0.004351	0.000174	0.000001
	25	0.814401	0.304826	0.037665	0.001649	0.000036	0.000001
	30	0.8366	0.287993	0.025734	0.000655	0.000004	0

誤差	T/c	0.4	0.5	0.6	0.7	0.8	0.9
	5	1.001664	1.006128	1.010461	1.015924	1.02454	1.02612
	10	1.002335	1.008033	1.01743	1.033082	1.039104	1.065004
	15	1.002436	1.01025	1.024914	1.040853	1.057687	0.979794
	20	1.002931	1.009751	1.025295	1.033829	1.009055	3.792053
	25	1.002528	1.012176	1.035089	1.059579	0.893986	0.279323
	30	1.002192	1.013709	1.039639	1.051733	1.499783	-

(4) ファクターモデルが Vasicek Model の数値実験の結果

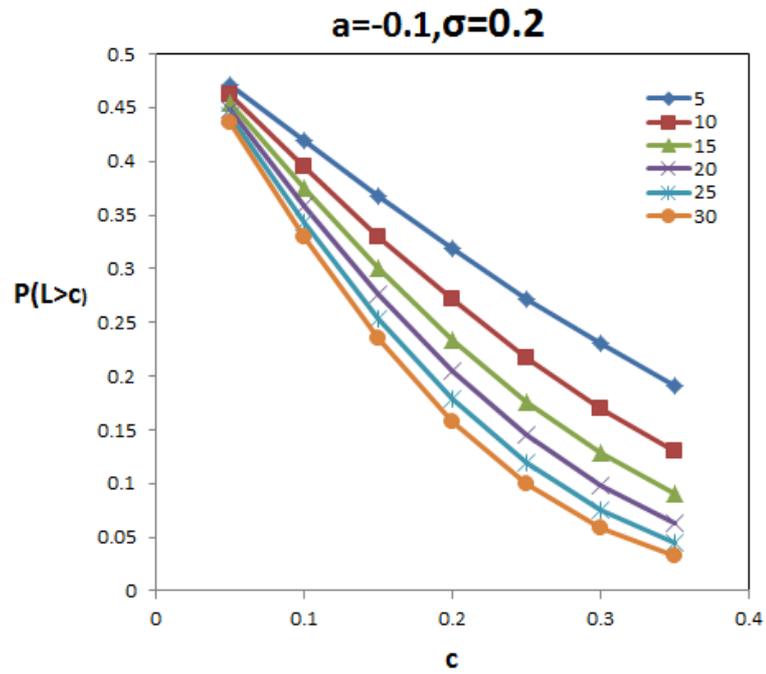


表 14 ファクターモデルが Vasicek Model である ( $a=-0.1, \sigma=0.2$ )

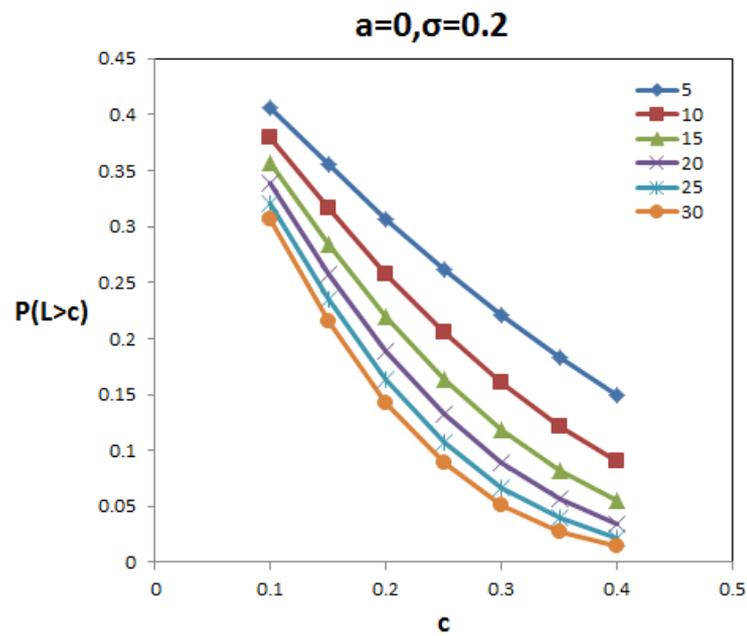


表 15 ファクターモデルが Vasicek Model である ( $a=0, \sigma=0.2$ )

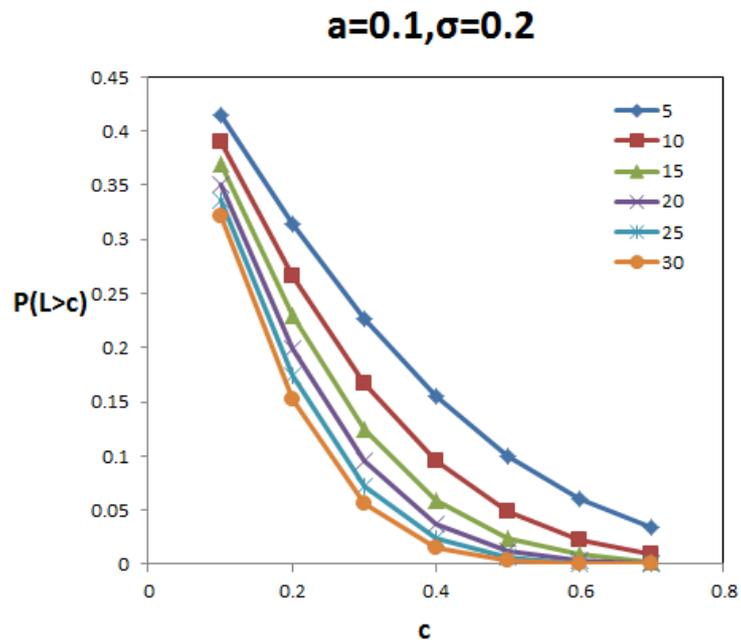


表 16 ファクターモデルが Vasicek Model である (a=0.1,σ=0.2)

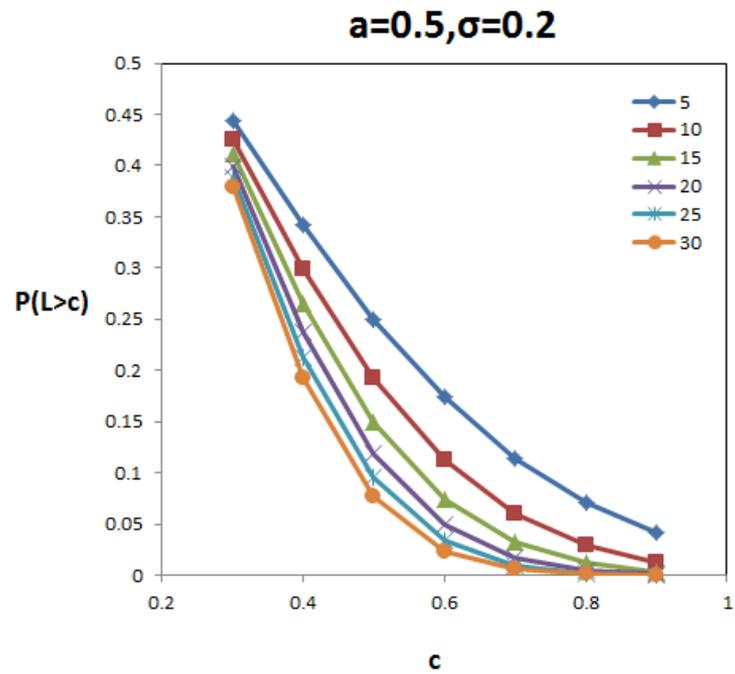


表 17 ファクターモデルが Vasicek Model である (a=0.5,σ=0.2)

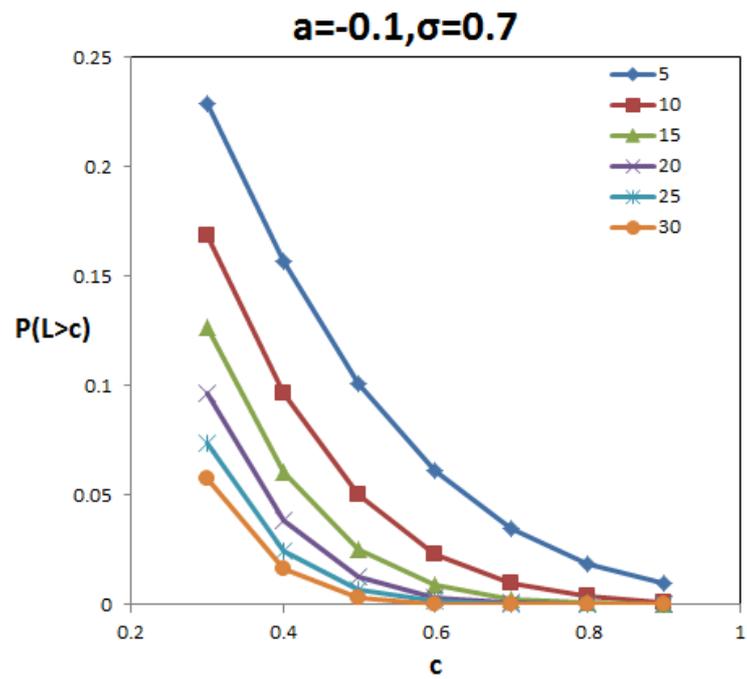


表 18 ファクターモデルが Vasicek Model である ( $a=-0.1, \sigma=0.7$ )

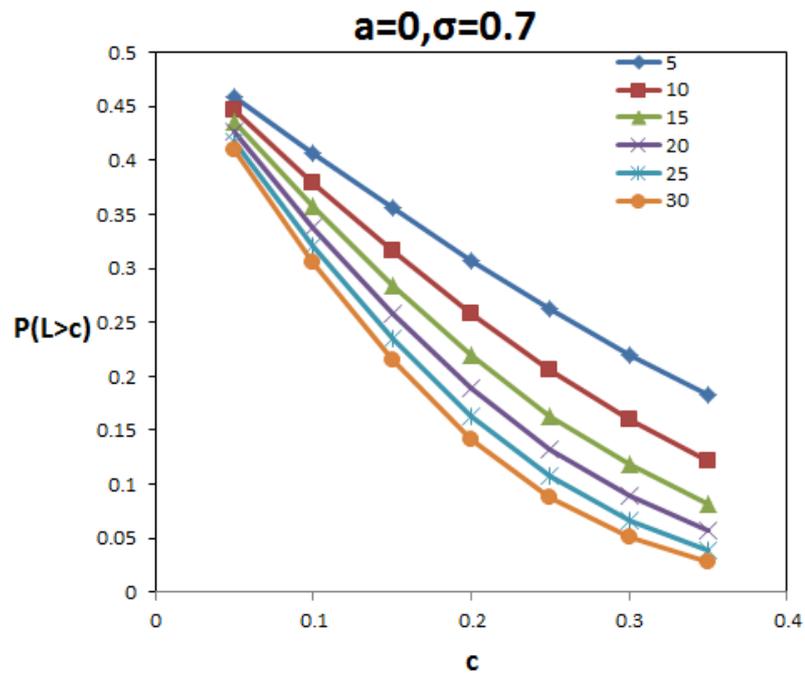


表 19 ファクターモデルが Vasicek Model である ( $a=0, \sigma=0.7$ )

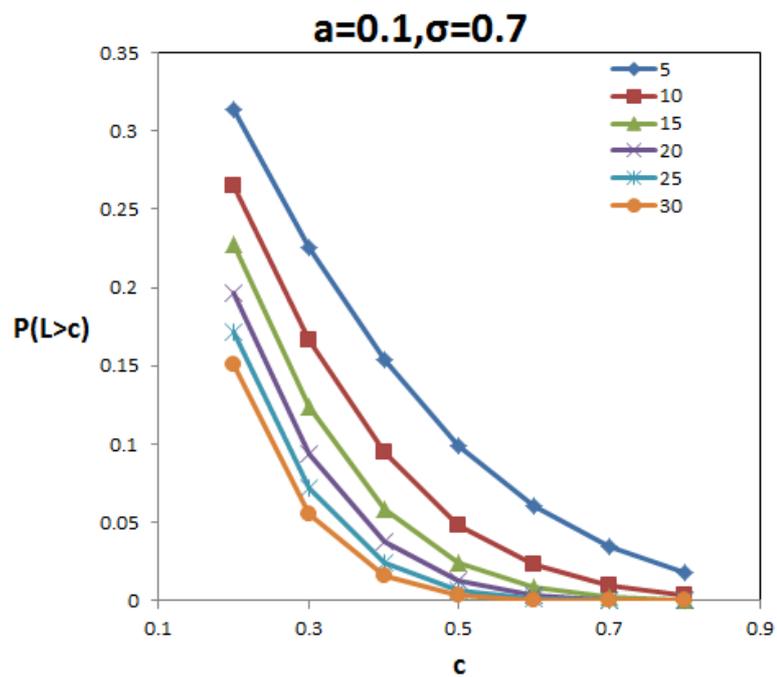


表 20 ファクターモデルが Vasicek Model である ( $a=0.1, \sigma=0.7$ )

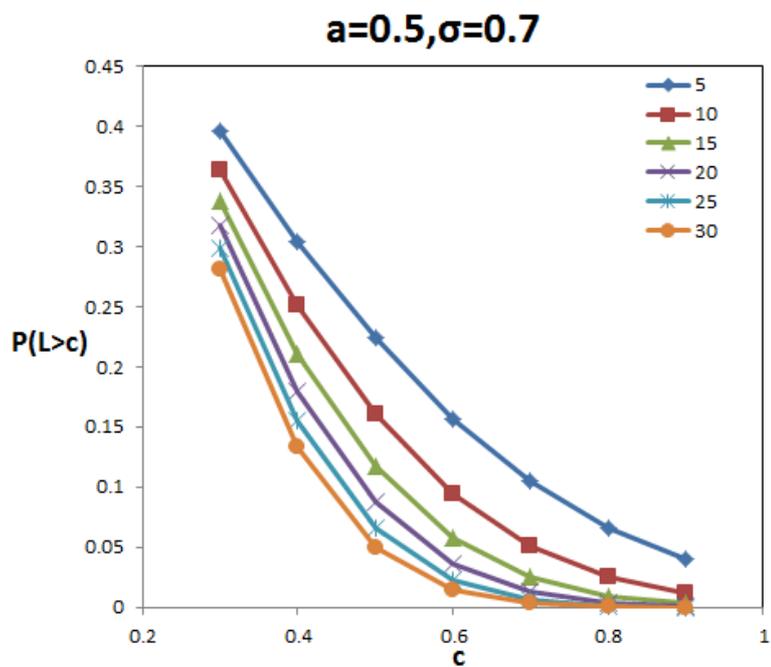


表 21 ファクターモデルが Vasicek Model である ( $a=0.5, \sigma=0.7$ )

#### (5) ファクターモデルが Vasicek Model の数値実験の結果についての考察

ファクターがバシチェックモデルの時、確率の理論的な値が得られない。そのため、企業の成長率  $a$  と株価のボラティリティ  $\sigma$  はブラック・ショールズの時と同じように設定し、数値実験を行った。二つのモデルのグラフを比較すると、裾の形がほぼ同じようになっている。つまり、 $c$  に対し、時間  $T$  が大きいほど、目標値  $c$  を上回る確率が小さくなる。Pham[1] の理論では  $P[L_T^a \geq c] \approx \exp(-\Gamma(c, \pi)T)$  で、確率  $P$  が指数関数に従い、時間  $T$  が大きくなるほど、確率が下がっていくことになる。従って、二つのモデルの実験値は Pham[1] の理論と整合性があるとみられる。

### 7. 付録

数値実験で使われた  $c$  言語のプログラムを掲載する。以下は Black-Scholes model の数値実験プログラムと Vasicek interest rate model の数値実験プログラムである。正規分布の乱数発生は Mersenne Twister を用いた。

#### (1) Black-Scholes model の数値実験プログラム

```
#define PI 3.14159265358979323846
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <time.h>
#include <string.h>

#define N 624
#define M 397
#define MATRIX_A 0x9908b0dfUL
#define UPPER_MASK 0x80000000UL
#define LOWER_MASK 0x7fffffffUL
static unsigned long mt[N];
static int mti=N+1;
//標準正規乱数発生 (Box-Muller)
double gasdev(void);
void init_genrand(unsigned long s)
{
    mt[0]= s & 0xffffffffUL;
    for (mti=1; mti<N; mti++) {
        mt[mti] =
            (1812433253UL * (mt[mti-1] ^ (mt[mti-1] >> 30)) + mti);
        mt[mti] ^= 0xffffffffUL;
    }
}
void init_by_array(unsigned long init_key[], int key_length)
{
    int i, j, k;
    init_genrand(19650218UL);
    i=1; j=0;
    k = (N>key_length ? N : key_length);
    for (; k; k--) {
        mt[i] = (mt[i] ^ ((mt[i-1] ^ (mt[i-1] >> 30)) * 1664525UL))
            + init_key[j] + j; /* non linear */
        mt[i] ^= 0xffffffffUL; /* for WORDSIZE > 32 machines */
        i++; j++;
        if (i>=N) { mt[0] = mt[N-1]; i=1; }
        if (j>=key_length) j=0;
    }
    for (k=N-1; k; k--) {
```

```

    mt[i] = (mt[i] ^ ((mt[i-1] ^ (mt[i-1] >> 30)) * 1566083941UL))
        - i; /* non linear */
    mt[i] &= 0xffffffffUL; /* for WORDSIZE > 32 machines */
    i++;
    if (i>=N) { mt[0] = mt[N-1]; i=1; }
}

mt[0] = 0x80000000UL; /* MSB is 1; assuring non-zero initial array */
}
unsigned long genrand_int32(void)
{
    unsigned long y;
    static unsigned long mag01[2]={0x0UL, MATRIX_A};
    if (mti >= N) {
        int kk;

        if (mti == N+1)
            init_genrand(5489UL);

        for (kk=0;kk<N-M;kk++) {
            y = (mt[kk]&UPPER_MASK)|(mt[kk+1]&LOWER_MASK);
            mt[kk] = mt[kk+M] ^ (y >> 1) ^ mag01[y & 0x1UL];
        }
        for (;kk<N-1;kk++) {
            y = (mt[kk]&UPPER_MASK)|(mt[kk+1]&LOWER_MASK);
            mt[kk] = mt[kk+(M-N)] ^ (y >> 1) ^ mag01[y & 0x1UL];
        }
        y = (mt[N-1]&UPPER_MASK)|(mt[0]&LOWER_MASK);
        mt[N-1] = mt[M-1] ^ (y >> 1) ^ mag01[y & 0x1UL];

        mti = 0;
    }
    y = mt[mti++];
    y ^= (y >> 11);
    y ^= (y << 7) & 0x9d2c5680UL;
    y ^= (y << 15) & 0xefc60000UL;
    y ^= (y >> 18);
    return y;
}
/* generates a random number on [0,1]-real-interval */
double genrand_real1(void)
{
    return genrand_int32()*(1.0/4294967295.0);
    /* divided by 2^32-1 */
}
/* generates a random number on [0,1)-real-interval */
double genrand_real2(void)
{
    return genrand_int32()*(1.0/4294967296.0);
    /* divided by 2^32 */
}
/* generates a random number on (0,1)-real-interval */
double genrand_real3(void)
{
    return (((double)genrand_int32()) + 0.5)*(1.0/4294967296.0);
    /* divided by 2^32 */
}

```

```

}
/* generates a random number on [0,1) with 53-bit resolution*/
double genrand_res53(void)
{
    unsigned long a=genrand_int32()>>5, b=genrand_int32()>>6;
    return(a*67108864.0+b)*(1.0/9007199254740992.0);
}
double gasdev(void){
static int iset=0;
static double gset;
double fac,rsq,v1,v2;

if (iset == 0){ //手持ちの乱数が無い
do {
v1=2.0*genrand_real3()-1.0; //各方向に - 1 から 1 まで広がる正方形の内部に
v2=2.0*genrand_real3()-1.0; //一様に点をとる
rsq=v1*v1+v2*v2; //単位円内かどうか調べ
}
while (rsq>=1.0 || rsq ==0.0); //そうでなければやり直し
fac=sqrt(-2.0*log(rsq)/rsq);
gset=v1*fac; //ボックスマラーの変換を使ってふたつの正規乱数を作る。一方を返し、もう一方を次回までに保存する。
iset=1;
return v2*fac;
}
else { //手持ちの乱数あり
iset=0;
return gset; //手持ちの乱数を返す
}
}

int main(void){
double Delta,r,sigma,a,t,b;
    int j,ll,t1,n;

double x[7600],z[7600],y[7600],e[7600],f[7600],c[7600],d[7600];
double l1,l2,l3,l4,l5,l6,l7;
double c1,c2,c3,c4,c5,c6,c7;
double count1,count2,count3,count4,count5,count6,count7;

Delta=0.004;//変更なし

r=0.01;
sigma=0.7;

t=5.0;
t1=5;//t=t1(t->double,t1->int)
n=t1*250;//n=t1*250(整数) 変えるとき、下も一緒に！

    a=0.1;//c0,c1,c2,a を変える！

c1=0.00406;
c2=0.2;
c3=0.3;

```

```

c4=0.4;
    c5=0.5;
c6=0.6;
c7=0.7;

FILE *portfolio;
portfolio=fopen("portfolio.xls","w");

x[1]=1.0;
z[1]=1.0;
y[1]=1.0;
e[1]=1.0;
f[1]=1.0;
c[1]=1.0;
d[1]=1.0;

count1=0.0;
count2=0.0;
count3=0.0;
count4=0.0;
count5=0.0;
count6=0.0;
count7=0.0;

for(ll=1;ll<=1000000;ll++){
for(j=1;j<=n;j++){

b=gasdev();
x[j+1] =x[j]+x[j]*((r+(sqrt(2.0*c1))*(a-r))*Delta+(sqrt(2.0*c1))*sigma*(sqrt(Delta))*b);
z[j+1] =z[j]+z[j]*((r+(sqrt(2.0*c2))*(a-r))*Delta+(sqrt(2.0*c2))*sigma*(sqrt(Delta))*b);
y[j+1] =y[j]+y[j]*((r+(sqrt(2.0*c3))*(a-r))*Delta+(sqrt(2.0*c3))*sigma*(sqrt(Delta))*b);
e[j+1] =e[j]+e[j]*((r+(sqrt(2.0*c4))*(a-r))*Delta+(sqrt(2.0*c4))*sigma*(sqrt(Delta))*b);
f[j+1] =f[j]+f[j]*((r+(sqrt(2.0*c5))*(a-r))*Delta+(sqrt(2.0*c5))*sigma*(sqrt(Delta))*b);
c[j+1] =c[j]+c[j]*((r+(sqrt(2.0*c6))*(a-r))*Delta+(sqrt(2.0*c6))*sigma*(sqrt(Delta))*b);
d[j+1] =d[j]+d[j]*((r+(sqrt(2.0*c7))*(a-r))*Delta+(sqrt(2.0*c7))*sigma*(sqrt(Delta))*b);

}

l1=log(x[n+1])/t;
l2=log(z[n+1])/t;
l3=log(y[n+1])/t;
l4=log(e[n+1])/t;
l5=log(f[n+1])/t;
l6=log(c[n+1])/t;
l7=log(d[n+1])/t;

if(l1>=c1) count1=count1+1.0;
if(l2>=c2) count2=count2+1.0;
if(l3>=c3) count3=count3+1.0;

```

```

if(l4>=c4) count4=count4+1.0;
if(l5>=c5) count5=count5+1.0;
if(l6>=c6) count6=count6+1.0;
if(l7>=c7) count7=count7+1.0;

```

```

}

```

```

count1=count1/1000000.0;
count2=count2/1000000.0;
count3=count3/1000000.0;
count4=count4/1000000.0;
count5=count5/1000000.0;
count6=count6/1000000.0;
count7=count7/1000000.0;

```

```

fprintf(portfolio,"%lf\n",count1);
fprintf(portfolio,"%lf\n",count2);
fprintf(portfolio,"%lf\n",count3);
fprintf(portfolio,"%lf\n",count4);
fprintf(portfolio,"%lf\n",count5);
fprintf(portfolio,"%lf\n",count6);
fprintf(portfolio,"%lf\n",count7);

```

```

} //main 終了の}

```

## (2)Vasicek interest rate model の数値実験プログラム

```

#define PI 3.14159265358979323846
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <time.h>
#include <string.h>

#define N 624
#define M 397
#define MATRIX_A 0x9908b0dfUL
#define UPPER_MASK 0x80000000UL
#define LOWER_MASK 0x7fffffffUL

static unsigned long mt[N];
static int mti=N+1;

//標準正規乱数発生 (Box-Muller)
double gasdev(void);

void init_genrand(unsigned long s)
{
    mt[0]= s & 0xffffffffUL;
    for (mti=1; mti<N; mti++) {
        mt[mti] =
            (1812433253UL * (mt[mti-1] ^ (mt[mti-1] >> 30)) + mti);

```

```

        mt[mti] &= 0xffffffffUL;
    }
}
void init_by_array(unsigned long init_key[], int key_length)
{
    int i, j, k;
    init_genrand(19650218UL);
    i=1; j=0;
    k = (N>key_length ? N : key_length);
    for (; k; k--) {
        mt[i] = (mt[i] ^ ((mt[i-1] ^ (mt[i-1] >> 30)) * 1664525UL))
            + init_key[j] + j; /* non linear */
        mt[i] &= 0xffffffffUL; /* for WORDSIZE > 32 machines */
        i++; j++;
        if (i>=N) { mt[0] = mt[N-1]; i=1; }
        if (j>=key_length) j=0;
    }
    for (k=N-1; k; k--) {
        mt[i] = (mt[i] ^ ((mt[i-1] ^ (mt[i-1] >> 30)) * 1566083941UL))
            - i; /* non linear */
        mt[i] &= 0xffffffffUL; /* for WORDSIZE > 32 machines */
        i++;
        if (i>=N) { mt[0] = mt[N-1]; i=1; }
    }

    mt[0] = 0x80000000UL; /* MSB is 1; assuring non-zero initial array */
}
unsigned long genrand_int32(void)
{
    unsigned long y;
    static unsigned long mag01[2]={0x0UL, MATRIX_A};
    if (mti >= N) {
        int kk;

        if (mti == N+1)
            init_genrand(5489UL);

        for (kk=0;kk<N-M;kk++) {
            y = (mt[kk]&UPPER_MASK)|(mt[kk+1]&LOWER_MASK);
            mt[kk] = mt[kk+M] ^ (y >> 1) ^ mag01[y & 0x1UL];
        }
        for (;kk<N-1;kk++) {
            y = (mt[kk]&UPPER_MASK)|(mt[kk+1]&LOWER_MASK);
            mt[kk] = mt[kk+(M-N)] ^ (y >> 1) ^ mag01[y & 0x1UL];
        }
        y = (mt[N-1]&UPPER_MASK)|(mt[0]&LOWER_MASK);
        mt[N-1] = mt[M-1] ^ (y >> 1) ^ mag01[y & 0x1UL];

        mti = 0;
    }
    y = mt[mti++];
    y ^= (y >> 11);
    y ^= (y << 7) & 0x9d2c5680UL;
    y ^= (y << 15) & 0xefc60000UL;
    y ^= (y >> 18);
    return y;
}

```

```

}
/* generates a random number on [0,1]-real-interval */
double genrand_real1(void)
{
    return genrand_int32()*(1.0/4294967295.0);
    /* divided by 2^32-1 */
}
/* generates a random number on [0,1)-real-interval */
double genrand_real2(void)
{
    return genrand_int32()*(1.0/4294967296.0);
    /* divided by 2^32 */
}
/* generates a random number on (0,1)-real-interval */
double genrand_real3(void)
{
    return (((double)genrand_int32()) + 0.5)*(1.0/4294967296.0);
    /* divided by 2^32 */
}
/* generates a random number on [0,1) with 53-bit resolution*/
double genrand_res53(void)
{
    unsigned long a=genrand_int32()>>5, b=genrand_int32()>>6;
    return(a*67108864.0+b)*(1.0/9007199254740992.0);
}
double gasdev(void){
static int iset=0;
static double gset;
double fac,rsq,v1,v2;

if (iset == 0){ //手持ちの乱数が無い
do {
v1=2.0*genrand_real3()-1.0; //各方向に - 1 から 1 まで広がる正方形の内部に
v2=2.0*genrand_real3()-1.0; //一様に点をとる
rsq=v1*v1+v2*v2; //単位円内かどうか調べ
}
while (rsq>=1.0 || rsq ==0.0); //そうでなければやり直し
fac=sqrt(-2.0*log(rsq)/rsq);
gset=v1*fac; //ボックスマラーの変換を使ってふたつの正規乱数を作る。一方を返し、もう一方を次回までに保存する。
iset=1;
return v2*fac;

}
else { //手持ちの乱数あり
iset=0;
return gset; //手持ちの乱数を返す
}
}
int main(void){
double Delta,r1,r2,r3,r4,r5,r6,r7,sigma,a,t,bx,by,b,k,y1,v,a0,b0,a1,b1,mu1,mu2,mu3,mu4,mu5,mu6,mu7,pi1,pi2,pi3,pi4,
int j,ll,t1,n;

double x[7600],y[7600],c[7600],cc[7600],d[7600],dd[7600],e[7600],ee[7600],f[7600],ff[7600],g[7600],gg[7600],h[7600]
double l1,l2,l3,l4,l5,l6,l7;
double c1,c2,c3,c4,c5,c6,c7;

```

```

double count1,count2,count3,count4,count5,count6,count7;

sigma=0.2;
Delta=0.004;

k=1.0;
y1=0.0;//y1=ybar
v=1.0;//金利モデル Y(t) の係数設定

a=0.1;
b=1.0;//mu(Yt)=a+b(Yt) 株の傾き, 株価に金利のノイズを乗せる

a0=0.01;//金利
b0=0.0;//r(Yt)=a0+b0(Yt) 金利の傾き

a1=a-a0;//a1=abar
b1=b-b0;//b1=bbar

t=5.0;//期間
t1=5;//t=t1(t->double,t1->int)
n=t1*250;//n=t1*250(整数) 変えるとき、下も一緒に！

c1=0.35;
c2=0.40;
c3=0.45;
c4=0.50;
c5=0.55;
c6=0.60;
c7=0.65;

lam=0.1;//ラムダの設定

FILE *vasicek;
vasicek=fopen("vasicek.xls","w");

count1=0.0;
count2=0.0;
count3=0.0;
count4=0.0;
count5=0.0;
count6=0.0;
count7=0.0;

y[1]=0.01;//金利モデル
x[1]=1.0;//ポートフォリオモデル

c[1]=0.01;//金利モデル
cc[1]=1.0;//ポートフォリオモデル

d[1]=0.01;
dd[1]=1.0;

e[1]=0.01;
ee[1]=1.0;

```

```

f[1]=0.01;
ff[1]=1.0;

g[1]=0.01;
gg[1]=1.0;

h[1]=0.01;
hh[1]=1.0;

for(ll=1;ll<=1000000;ll++){

for(j=1;j<=n;j++){
    by=gasdev();
    bx=gasdev();

r1=a0+b0*y[j]; //金利 r の関数
r2=a0+b0*c[j];
r3=a0+b0*d[j];
r4=a0+b0*e[j];
r5=a0+b0*f[j];
r6=a0+b0*g[j];
r7=a0+b0*h[j];

mu1=a1+b1*y[j]; //mubar
mu2=a1+b1*c[j];
mu3=a1+b1*d[j];
mu4=a1+b1*e[j];
mu5=a1+b1*f[j];
mu6=a1+b1*g[j];
mu7=a1+b1*h[j];

pi1=(b1*y[j])/(1.0-lam*(c1))+a1;
pi2=(b1*c[j])/(1.0-lam*(c2))+a1;
pi3=(b1*d[j])/(1.0-lam*(c3))+a1;
pi4=(b1*e[j])/(1.0-lam*(c4))+a1;
pi5=(b1*f[j])/(1.0-lam*(c5))+a1;
pi6=(b1*g[j])/(1.0-lam*(c6))+a1;
pi7=(b1*h[j])/(1.0-lam*(c7))+a1;

x[j+1] =x[j]+x[j]*((r1+pi1*mu1)*Delta+pi1*sigma*(sqrt(Delta))*bx); //ポートフォリオを求める
cc[j+1] =cc[j]+cc[j]*((r2+pi2*mu2)*Delta+pi2*sigma*(sqrt(Delta))*bx);
dd[j+1] =dd[j]+dd[j]*((r3+pi3*mu3)*Delta+pi3*sigma*(sqrt(Delta))*bx);
ee[j+1] =ee[j]+ee[j]*((r4+pi4*mu4)*Delta+pi4*sigma*(sqrt(Delta))*bx);
ff[j+1] =ff[j]+ff[j]*((r5+pi5*mu5)*Delta+pi5*sigma*(sqrt(Delta))*bx);
gg[j+1] =gg[j]+gg[j]*((r6+pi6*mu6)*Delta+pi6*sigma*(sqrt(Delta))*bx);
hh[j+1] =hh[j]+hh[j]*((r7+pi7*mu7)*Delta+pi7*sigma*(sqrt(Delta))*bx);

y[j+1]=y[j]-k*(y[j]-y1)*Delta + v*(sqrt(Delta))*by; //金利モデル Yt を求める
c[j+1]=c[j]-k*(c[j]-y1)*Delta + v*(sqrt(Delta))*by;
d[j+1]=d[j]-k*(d[j]-y1)*Delta + v*(sqrt(Delta))*by;
e[j+1]=e[j]-k*(e[j]-y1)*Delta + v*(sqrt(Delta))*by;
f[j+1]=f[j]-k*(f[j]-y1)*Delta + v*(sqrt(Delta))*by;

```

```

g[j+1]=g[j]-k*(g[j]-y1)*Delta + v*(sqrt(Delta))*by;
h[j+1]=h[j]-k*(h[j]-y1)*Delta + v*(sqrt(Delta))*by;

}

l1=log(x[n+1])/t;
l2=log(cc[n+1])/t;
l3=log(dd[n+1])/t;
l4=log(ee[n+1])/t;
l5=log(ff[n+1])/t;
l6=log(gg[n+1])/t;
l7=log(hh[n+1])/t;

if(l1>=c1) count1=count1+1.0;
if(l2>=c2) count2=count2+1.0;
if(l3>=c3) count3=count3+1.0;
if(l4>=c4) count4=count4+1.0;
if(l5>=c5) count5=count5+1.0;
if(l6>=c6) count6=count6+1.0;
if(l7>=c7) count7=count7+1.0;

}

count1=count1/1000000.0;
count2=count2/1000000.0;
count3=count3/1000000.0;
count4=count4/1000000.0;
count5=count5/1000000.0;
count6=count6/1000000.0;
count7=count7/1000000.0;

fprintf(vasicek,"%lf\n",count1);
fprintf(vasicek,"%lf\n",count2);
fprintf(vasicek,"%lf\n",count3);
fprintf(vasicek,"%lf\n",count4);
fprintf(vasicek,"%lf\n",count5);
fprintf(vasicek,"%lf\n",count6);
fprintf(vasicek,"%lf\n",count7);

} //main 終了の}

```

#### 参考文献

- [1] Huyen Pham.: A large deviations approach to optimal long term investment *Finance and Stochastics*, (7):169-175, 2003.
- [2] Dembo,A.,Zeitouni,O,: Large deviations techniques and applications,2nd Ed. *Berlin Heidelberg,New York Springer* (1998).
- [3] Rockafellar,R.T.: Convex Analysis. *Princeton University Press* 1970.
- [4] H.P.William: ニューメリカルレシビ・イン・シー日本語版-C 言語による数値計算のレシビ, 技術評論社 (1993/06).
- [5] D.Lamberton,B.Lampeyre: ファイナンスへの確率解析, 朝倉書店 (2000/06).
- [6] Mersenne Twister(<http://www.natural-science.or.jp>)

## 8. 謝辞

本論文の作成にあたり、長きに渡りご指導いただいた浦谷規教授に、深く感謝いたします。また、素晴らしい研究環境を与えて下さったことにも感謝いたします。また、研究面で様々な助言と指導を頂いた安田和弘助教授にも、感謝いたします。そして、多くの助力を頂いた下谷健太と問谷昭博と小坂和哉にも感謝をいたします。最後に、学生生活を支えてくれた家族に感謝いたします。

2013年2月

陳 瑶瑶