観測不可な確率変数に関する推計とファイナンス工学への応用

三田 光星

概要

この論文は特定のモデルにおいて観測不可な確率変数を推計し, ファイナンス工学へ応用した結果をまとめたものである. 経済学において観測不可な変数は頻繁に登場するが, その多くが仮説に大きく寄与する変数でありながら, 扱いが非常に難しいものとされる. たとえば, 計量経済学では欠測値と相関しない変数を使って回帰分析を行う操作変数法がよく知られているが, 肝心の操作変数を見つけることは困難であることも多い. しかし, そのような変数についての情報を得ることは, 多くの示唆を得ることにつながる. したがって, この論文では確率過程論をベースに観測不可な確率変数の推計とその応用について考える.

第 2 章では年度末に株式市場で見られる株価の過熱をモデル化し, 配当の影響を受けなかった場合である理論上の株価の動きを推計する. この株価の過熱は配当によるものと考えられ, 配当を受け取る権利が確定した後, 大きく下落する形で過熱前の元の水準に戻るような動きが株式市場では頻繁に見られる. 将来キャッシュフローの現在価値として株価は表現されるため, 通常は起こり得ないと考えられるこの現象をレジームスイッチングモデルとカルマンフィルターを用いて捉える手法について考える. そしてそれらの手法によって配当による過熱が引き起こされない場合の株価を事前に推計する.

第 3 章では災害債券のリスクを実際の被害のデータから推計するのではなく, 債券の価格データから推計する逆問題について考える. 災害債券は災害による被害の総額などの指標 A_t に基づいて価値が決まる金融商品であり, そのリスクの性質から市場リスクがその他の資産と比較して非常に小さい債券であることが知られている. 通常は過去の災害の被害状況のデータを用いて指標 A_t の動きを推計し, 金利に CIR モデルを適用するなど特定のモデルによって債券を評価するという手法が用いられるが, 被害状況のデータが常に正確に得られるかは定かではなく, 観測不可な場合もあると考えられる. そのような観測不可な指標 A_t について, 誘導型モデルと実際の債券の価格データから数値計算によって推計し, 複雑な構造を持った災害債券の評価に応用する.

最後に第 4 章では特定の領域の最終脱出時刻 (以下, Last Exit Time) とその領域の滞在時間 (以下, Occupation Time) の同時分布をラプラス変換の形で求め, その分布を用いた費用最小化問題の解法について考える. Last Exit Time は通常のフィルトレーションにおいて可測ではなく, 特定の領域を脱出した時刻 t において, 真に Last Exit Time が t であるかは実際に確率過程が止まるまで判別ができない. 現在時刻においては観測できない変数である Last Exit Time について同時分布を求めることで, 可測である Occupation Time の情報を用いた条件付きの分布を考えることができ, より正確に Last Exit Time の検討が可能となる.

謝辞

本論文は筆者が京都大学大学院経済学研究科博士後期課程に在籍中の研究成果をまとめたものである. 同研究科教授 江上 雅彦 先生には指導教官として本研究の実施の機会を与えていただき, その遂行にあたって終始ご指導をいただいた. ここに深謝の意を表する. 同大学経済研究所教授 西山 慶彦 先生には副指導教官としてご助言とご指導をいただいた. ここに深謝の意を表する. また同研究科博士で講師であるケヴヘイッシュウィリルースダン先生や柳 貴英先生をはじめ, 同研究科・経済学部所属の江上ゼミ生には研究遂行にあたり日頃より有益なご助言をいただいた. ここに感謝の意を表する.

なお, 本研究の一部は日本学術振興会特別研究員奨励費 (課題番号 No. 20J15618) によった.

最後にこの論文の完成にあたって経済的に支えてくれた家族と, 趣味での交流を通して精神的に支えてくれた友人たちと全てのベニュー, ポップカルチャーとサブカルチャーに感謝の意を表して謝辞とさせていただく.

目次

第 1 章

Introduction

三田 光星

この論文は特定のモデルにおいて観測不可な確率変数を推計し, ファイナンス工学へ応用した結果をまとめたものである. 経済学において観測不可な変数は頻繁に登場するが, その多くが仮説に大きく寄与する変数でありながら, 扱いが非常に難しいものとされる. たとえば, 計量経済学では欠測値と相関しない変数を使って回帰分析を行う操作変数法がよく知られているが, 肝心の操作変数を見つけることは困難であることも多い. しかし, そのような変数についての情報を得ることは, 多くの示唆を得ることにつながる. したがって, この論文では確率過程論をベースに観測不可な確率変数の推計とその応用について考える.

第 2 章では年度末等に株式市場で見られる株価の過熱をモデル化し, 配当の影響を受けなかった場合である理論上の株価の動きを推計する. この株価の過熱は配当によるものと考えられ, 配当を受け取る権利が確定した後, 大きく下落する形で過熱前の元の水準に戻るような動きが株式市場では頻繁に見られる. 将来キャッシュフローの現在価値として株価は表現されるため, 通常は起こり得ないと考えられるこの現象をレジームスイッチングモデルとカルマンフィルターを用いて捉える手法について考え, 配当による過熱が引き起こされない場合の株価を推計する.

レジームスイッチングモデルは観測不可な状態変数に応じてパラメータや設定が転換するモデルである. 第 2 章においては平常時と株価過熱時の 2 つの状態を考え, それぞれの設定におけるパラメータを推計する. 各状態確率も同時に計算されるため, 実際に時間が経過するうちに過熱状態に株式市場が入るかどうかの確認や, 転換の時点を閾値に応じて決定することも可能となる.

一方でカルマンフィルターに代表されるフィルタリングは観測可能なデータと関係がある確率過程の推計において有効な手法である. 特にカルマンフィルターは観測可能なデータと線形の関係式を持つ確率過程を推計する方法であり, 単純な形でありながら有用性が高い. 第 2 章においては過熱時の株価を観測可能なデータ, 配当の権利確定後の株価に繋がる理論上の株価を観測不可な確率過程としてフィルタリングを行う. また, これらの推計手法を応用し, 配当確定後の株価をシミュレーションを用いて事前に推測することを考える.

第 3 章では災害債券のリスクを実際の被害のデータから推計するのではなく, 債券の価格データから推計する逆問題について考える. 災害債券は災害による被害の総額などの指標 A_t に基づいて価値が決まる金融商品であり, そのリスクの性質から市場リスクがその他の資産と比較して非常に小さい債券であることが知られている. Burnecki (2005) に代表される災害債券の評価モデルは, 過去の災害の被害状況のデータを用いて指標 A_t の動きを推計し, 金利モデルとして CIR モデルを採用するなど特定のモデルを適用することで債券を評価する. しかし, 被害状況のデータが常に正確に得られるかは定かではなく, 破壊的な被害によって観測が不可な場合も

あると考えられる. 加えて, 発行者である保険会社等と投資家の間には災害に関する情報量において差が存在していることも容易に想像できる. したがって, A_t を観測不可な確率過程として, 誘導型モデルと実際の債券の価格データから数値計算によって推計, 複雑な構造を持った災害債券の評価に応用する.

最後に第 4 章では確率過程が特定の領域を最後に脱出した時刻 (以下, Last Exit Time) とその領域に確率過程が滞在していた時間 (以下, Occupation Time) の同時分布をラプラス変換の形で求め, その分布を用いた費用最小化問題の解法について考える. Last Exit Time は特定の状態を脱出した時刻と考えることができ, 景気判断指標などさまざまな応用例を考えることができる. しかし, Last Exit Time は通常のフィルトレーションにおいて可測ではなく, 特定の領域を脱出した時刻 t において, 真に Last Exit Time が時刻 t であるかは実際に確率過程が kill されるまで判別ができない. 現在時刻においては観測できない変数である Last Exit Time について同時分布を求めることで, 可測である Occupation Time の情報を用いた条件付きの分布を考えることができ, より正確に Last Exit Time の検討が可能となる.

References

Burnecki, Krzysztof (2005). "Pricing catastrophe bonds in a compound non-homogeneous Poisson model with left-truncated loss distributions". In: *Mathematics in Finance Conference*.

Hamilton, James D. (1990). "Analysis of time series subject to changes in regime". In: *Journal of Econometrics* 45.1, pp. 39–70. ISSN: 0304-4076. DOI: https://doi.org/10.1016/0304-4076(90)90093-9. URL: https://www.sciencedirect.com/science/article/pii/0304407690900939.

Kim, Chang-Jin, Charles R Nelson, et al. (1999). "State-space models with regime switching: classical and Gibbs-sampling approaches with applications". In: *MIT Press Books* 1.

Tsay, Ruey S (2010). *Analysis of Financial Time Series*. John Wiley & Sons.

小西貞則, 越智義道, and 大森裕浩 (2008). *計算統計学の方法: ブートストラップ・EM アルゴリズム・MCMC*. シリーズ予測と発見の科学 / 北川源四郎, 有川節夫, 小西貞則, 宮野悟編. 朝倉書店. ISBN: 9784254127850. URL: https://books.google.co.jp/books?id=FIf1PAAACAAJ.

第 2 章

フィルタリングによる配当確定後株価の推計

三田 光星

2.1 Introduction

株価は理論的に割引キャッシュフロー法 (DCF 法) と呼ばれる方法を用いて, 将来キャッシュフローの現在価値として評価される. DCF 法においては配当の支払いが株価として最初から織り込み済みであり, 配当を得る権利が各時点で確定するかどうかによって価値が変動するということはないと考えられる.

しかし実際の市場においては, 期末が近づくにつれ株価が不自然に上昇し, 配当の権利が確定する権利付最終日後に大幅に下落する銘柄が多く存在する. 特に最低投資額が低く配当の高い「高配当銘柄」とされるものはこの傾向が強く, 権利確定日前後は不安定な変動が見られる. 要因として考えられるものは「配当狙い」という戦略を取る投資家の存在や, 「まだ間に合う, 高配当銘柄」といったタイトルで配当狙いの戦略を煽るようなニュース記事などが挙げられる. [*1]

この論文では上記のような不自然な株価の過熱を「投資家の当期配当に対する過剰な期待」によるものと考える. その動きを単純な状態空間を設定したレジームスイッチングモデルによって捉え, 配当確定直後の株価に至る観測不可なパスをカルマンフィルターによって推計することで, 当期配当が年度末付近の価格に強い影響を与えていることを検証することを目的とする.

このような分析からはいくつかの知見が得られる. 1 つは推計した観測不可なパスが企業価値過程を示唆するという点である. 本来株価と企業価値過程は密接な関係にあるが, 年度末が近づくにつれて株価が過熱する現象はあくまで当期配当が原因の市場性によるものであるため, 企業価値自体は変わらず推移し続けると考えられる. したがって観測不可なパスとして本来あるべき株価の動きを推計することで, 正しく企業価値評価を行うことが可能になる.

もう 1 つは配当を意識し始める時期も同時に推計されるという点である. レジームスイッチングモデルによる分析の過程で各時点での状態確率を推計することになるが, この状態確率をもとに投資家がどの時点から配当を意識し始めるかを検討することができ, 効率的な投資戦略を構築することが可能になると考えられる.

当論文の構成は以下のとおりである. 第 2 節では上記の仮説について説明し, その仮説を検証するための仮

[*1] たとえば株探では「【特集】まだ間に合う, 9 月配当【高利回り】ベスト 30「プライム」編 割安株特集」といったタイトルの記事が毎月末リリースされている

定と使用データについて記述する. 第 3 節では第 2 節で記述した仮説に基づいて手法を確立した上で, 実際に分析を行いその結果の考察を行う. 最後に第 4 節では事後的な分析を行った第 3 節の手法により, 期間途中のデータとシミュレーションを用いて配当確定直後の株価を事前に推計することを試みる.

2.2　仮説とデータの概要

2.2.1　仮説とそれに伴う仮定について

当論文では以下の仮説をもとに分析を行うものとする.

> 比較的少額から投資が可能な個別株について, 将来キャッシュフローの一部として株価に織り込み済みであるはずの当期配当が期間の途中で強く意識され, 株価のドリフトが変化する. そして配当を受け取る権利が確定した直後, 株価の過熱が下方ジャンプによって調整される.

この仮説をもとに検証を行う上での仮定について述べる. 確率空間 $(\Omega, \mathcal{F}, \mathbb{P})$ において株価は以下のような幾何ブラウン運動に従うこととする.

$$\frac{dX_t}{X_t} = \mu dt + \sigma dW_t$$
$$\Longleftrightarrow X_t = X_0 \exp\left[\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t\right]$$

なお, フィルトレーションは $\{\mathcal{F}_t\}$ で表す. このとき $(X_t)_{t\in\mathbb{R}_+}$ の対数をとった過程 $Y_t := \log(X_t)$ はドリフト付きブラウン運動であり, $r := \left(\mu - \frac{\sigma^2}{2}\right)$ とすれば

$$dY_t = rdt + \sigma dW_t \tag{2.1}$$

である. 以下では特に断りのない限り, 対数系列 $\{Y_t\}$ を用いて分析を行う. なお Merton (1976) に代表される, ジャンプの伴う拡散過程を株価モデルとして設定する場合もあるが, この場合権利落ち日というジャンプする時間が決まっており, 加えて高々年に 2 度であるため, ジャンプを捉えるという観点においては適切なモデルとは言い難い. したがって今回は数理ファイナンスや金融工学等において基本の設定である幾何ブラウン運動を用いている.

加えて状態空間における状態の数は 2 つであると仮定し, 状態変数を $\{S_t\}$, 配当を意識する前後という状態を $I = \{0, 1\}$ として設定する. 0 時点においては必ず状態 0 にいる, つまり $S_0 = 0$ とし, 各状態のパラメータについては添字 i をつけて表現することとする. 状態問わずボラティリティ σ は一定で, 状態 1 のドリフト μ_1 は状態 0 のドリフト μ_0 より大きいこととする. また, 投資家は配当額 D について事前に知っている, あるいは予測ができると仮定する. これは前年度の配当額やアナリストの予想等により一定の確度で知ることが可能であるため, 実際の状況と整合する仮定である.

2.2.2　利用データ

使用データはハードオフ (TYO: 2674), ゆうちょ銀行 (TYO: 7182), 旭化成 (TYO: 3407), 日東工業 (TYO: 6651) の 4 社の株価である. 業種は異なるが, いずれも最低投資金額が 2020 年度開始時点で 20 万円以下であり, 比較的少額から投資が可能な銘柄である. このうちハードオフとゆうちょ銀行は 3 月末の権利確定日の情報をもとに配当が支払われる期末一括配当の銘柄であり, 旭化成と日東工業は第 2 四半期と第 4 四半期の権利

確定日の情報をもとにそれぞれ中間配当と期末配当が支払われる銘柄である. 今回は当期配当が影響を与えている可能性について分析するため, 配当の支払われ方に応じて分析期間やデータの使用を以下の通りとした.

期末一括配当銘柄に関して

期末一括配当銘柄であるハードオフとゆうちょ銀行に関しては 1 年おきに配当が支払われることから, 分析期間を 1 年とすることが妥当である. したがって以下では 2020 年度と 2021 年度の 2 年間について分析を行うこととする. なお, 権利付最終日後しばらくは下落の影響で株価の動きが不安定になる可能性があるため, 第 1 四半期の終わりである 6 月末から権利付最終日までのデータを用いることとする.

銘柄	年度	配当	終値 (権利付最終日)	始値 (権利落ち日)	騰落率 (前日比)
ゆうちょ銀行	2020	50	1131	1080	−4.4209 %
	2021	50	1062	1010	-4.7081 %
ハードオフ	2020	35	898	851	−5.2339 %
	2021	40	806	766	−4.9628 %

表 2.1　配当と権利確定前後の下落について

中間配当ありの銘柄に関して

中間配当ありの銘柄である旭化成と日東工業については半年おきに配当が支払われることから, 分析期間を半年とすることが妥当である. したがって以下では 2021 年度の第 1・2 四半期と第 3・4 四半期について分析を行うこととする. 期末一括配当銘柄と同様, 権利落ち日の下落の影響で株価の動きが不安定になる可能性があるが, 分析期間の幅が半年となるため, 期末一括配当銘柄と同様の処理を行うとデータ数が不足する. したがって分析のための日次データの数を確保するために, 第 1 四半期, 第 3 四半期の期首 15 営業日を使用しないこととした.

銘柄	期間	配当	終値 (権利付最終日)	始値 (権利落ち日)	騰落率 (前日比)
旭化成	上半期	18	1131	1080	−4.4209 %
	下半期	18	1062	1010	−4.7081 %
日東工業	上半期	25	898	851	−5.2339 %
	下半期	25	806	766	−4.9628 %

表 2.2　配当と権利確定前後の下落について

2.3　検証手法の概要と結果

本節は前節における検証手法について確認し, 前述のデータを用いて分析, 結果を検討する.

以下では 2 段階で仮説の検証を行う. まず, ドリフトの異なる 2 つの状態 $I = \{0, 1\}$ について, レジームスイッチングモデルの手法を用いることでパラメータや状態確率の推計を行う. 推計の結果が

1. $\mu_0 < \mu_1$

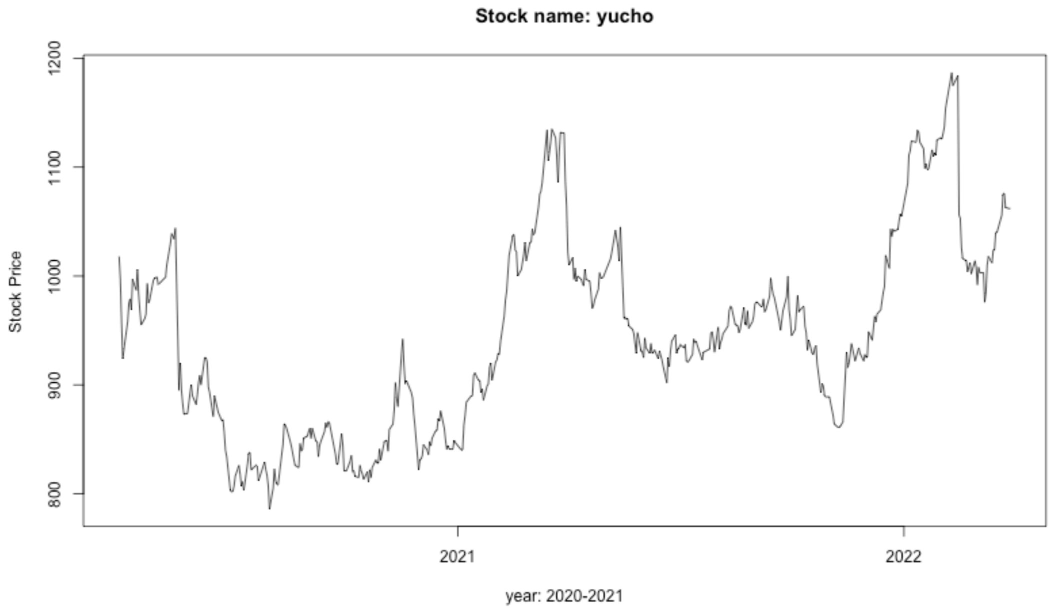

図 2.1　株価: ゆうちょ銀行

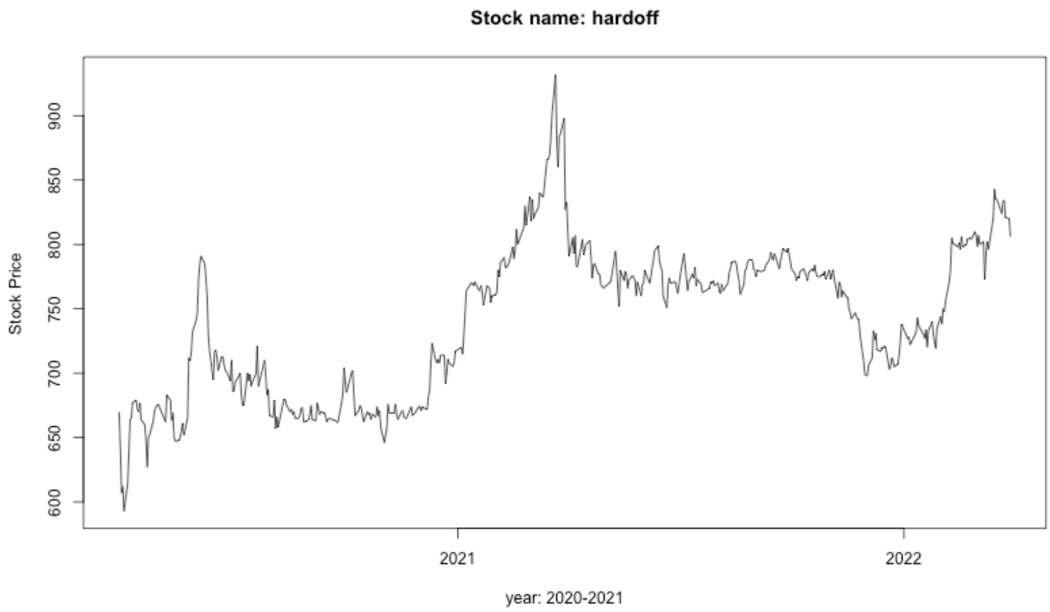

図 2.2　株価: ハードオフ

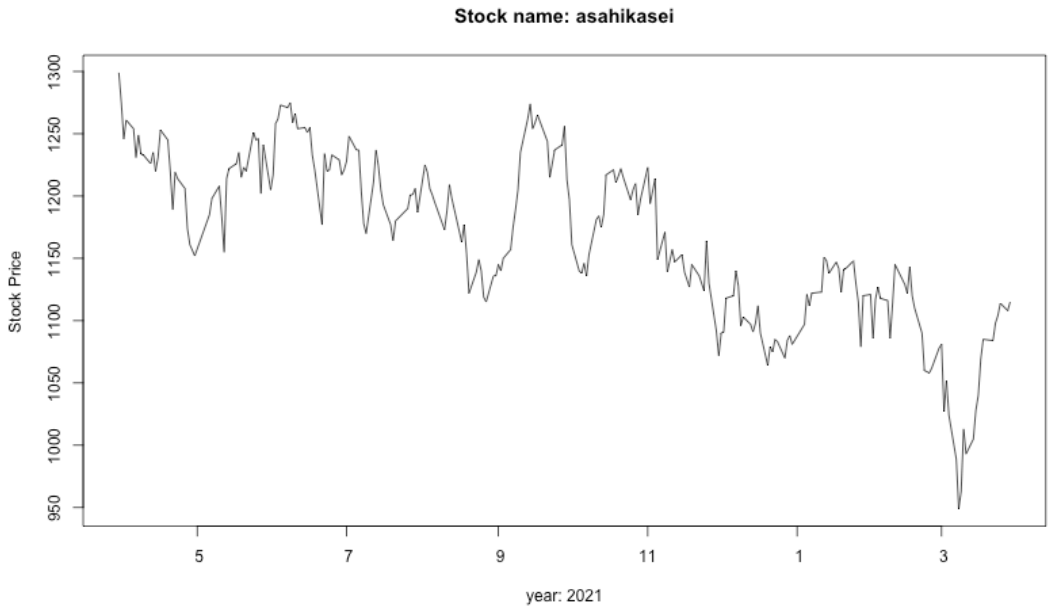

図 2.3　株価: 旭化成

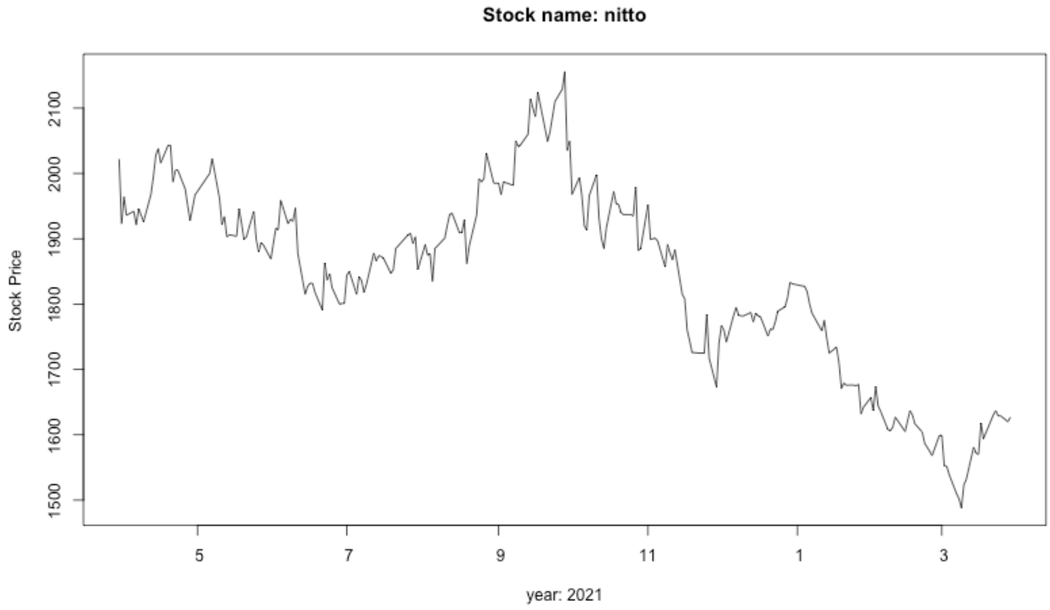

図 2.4　株価: 日東工業

2. $S_t = 1$ の状態確率が時間経過によって上昇する

という 2 つの条件を満たし, 仮説を支持していることを確認したのちに, 1 年間のデータと推計された状態確率から, 状態 1 の期間における理論上の株価のパス $\{\alpha_t\}$ をカルマンフィルターによって推定する. その端点 α_T が下落直後の価格である権利落ち日の始値とどの程度整合しているかを確認する.

2.3.1 レジームスイッチングモデル

本小節では検証の第 1 段階であるレジームスイッチングモデルの設定と前述のデータを用いた結果について記述する. 株価の確率微分方程式は (2.1) であり, 初期状態確率は時刻 0 において必ず状態 0 にいると仮定しているため $\mathbb{P}(S_0 = 0) = 1$, 推移確率行列 P を状態 i から j に移動する確率を p_{ij} として

$$P = \begin{bmatrix} 1 - p_{01} & p_{10} \\ p_{01} & 1 - p_{10} \end{bmatrix}$$

と設定する. したがって推計するパラメータは $(\sigma, \mu_0, \mu_1, p_{01}, p_{10})$ の 5 つである. パラメータの推計には EM アルゴリズムを用いた. レジームスイッチングモデルと EM アルゴリズムの詳細については付録 A を参照のこと.

期末一括配当

期末一括配当銘柄について上記の検証を行った結果は以下の表 2.3 と図 2.5〜2.8 である. 表 2.3 は EM アルゴリズムによって推計されたパラメータの値と標準誤差であり, 図 2.5〜2.8 は各時点において確率過程 $\{Y_t\}$ が状態 1 にいる確率 $P(S_t = 1)$ をプロットしたものである.

状態 1 の状態確率は単調増加であり, また仮定である $\mu_1 > \mu_0$ を満たしていることから, 何らかの要因で権利確定日に向けてドリフトが大きくなっていることが数値的に確認でき, 仮説を支持していると考えられる. ここで他と比べるとドリフトの推定値の標準誤差が少し大きいが. これは 2 つの状態を考慮するにあたって使用データの数が 180 程度と少ないことから発生していると考えられる. また, 状態確率を見ると 0.5 を超えるのは 10 月から 12 月にかけてであり, この時期から投資家が徐々に期末の配当を意識し始めると捉えることもできる. この時期は中間決算が発表される時期であり, その結果を確認した投資家が該当の銘柄を保有する姿勢を見せるために過熱が始まると考えることができる.

企業名	年度		σ	μ_0	μ_1
ゆうちょ銀行	2020 年度	推計値	0.2245845	0.2667084	0.6915573
		標準誤差	2.24009E-05	0.026974767	0.027532539
	2021 年度	推計値	0.2602232	0.1738067	0.23042
		標準誤差	2.73433E-05	0.026897782	0.039923104
ハードオフ	2020 年度	推計値	0.2561547	-0.02359164	0.5780303
		標準誤差	2.61702E-05	0.026904221	0.038091237
	2021 年度	推計値	0.1818151	0.04439213	0.1351205
		標準誤差	1.32647E-05	0.02341415	0.020882025

企業名	年度		p_{01}	p_{10}
ゆうちょ銀行	2020 年度	推計値	5.97176E-13	0.01004683
		標準誤差	6.77997E-09	0.000781578
	2021 年度	推計値	9.70393E-06	0.0163673
		標準誤差	2.07802E-05	0.001243641
ハードオフ	2020 年度	推計値	1.02498E-13	0.01587911
		標準誤差	2.63389E-09	0.00118609
	2021 年度	推計値	1.30654E-11	0.007258337
		標準誤差	3.07151E-08	0.000623839

表 2.3　推計結果 (年率). μ_i については r_i の標準誤差を用いている.

中間配当あり

中間配当がある銘柄について上記の検証を行った結果は以下の表 2.4 と図 2.9〜2.12 である. 期末一括配当銘柄と同様に, 表 2.4 は EM アルゴリズムによって推計されたパラメータの値と標準誤差であり, 図 2.9〜2.12 は各時点において確率過程 $\{Y_t\}$ が状態 1 にいる確率 $\mathbb{P}(S_t = 1)$ をプロットしたものである.

期末一括配当銘柄と同じく, 状態 1 の状態確率は単調増加であり, また仮定である $\mu_1 > \mu_0$ も十分に満たしていることから, 何らかの要因で権利確定日に向けてドリフトが大きくなっていることが確認され, 仮説を支持している. また状態確率を見ると 0.5 を超えるのは上半期は 6 月から 7 月にかけて, 下半期は 11 月から 12 月にかけてであり, この時期から投資家が徐々に各期末の配当を意識し始めると捉えることもできる. これはそれぞれ決算の発表時期であり, 期末一括配当と同様の理由であると考えることができる. 加えて, 上半期より下半期の方が状態確率の上昇が早いことも確認される. これは上半期終了時の配当支払いは一部銘柄に限られるが, 下半期終了時は期末一括配当銘柄も配当の支払いを行うため, 投資家の間で上半期より下半期の方が配当を強く意識するからではないかと考えられる.

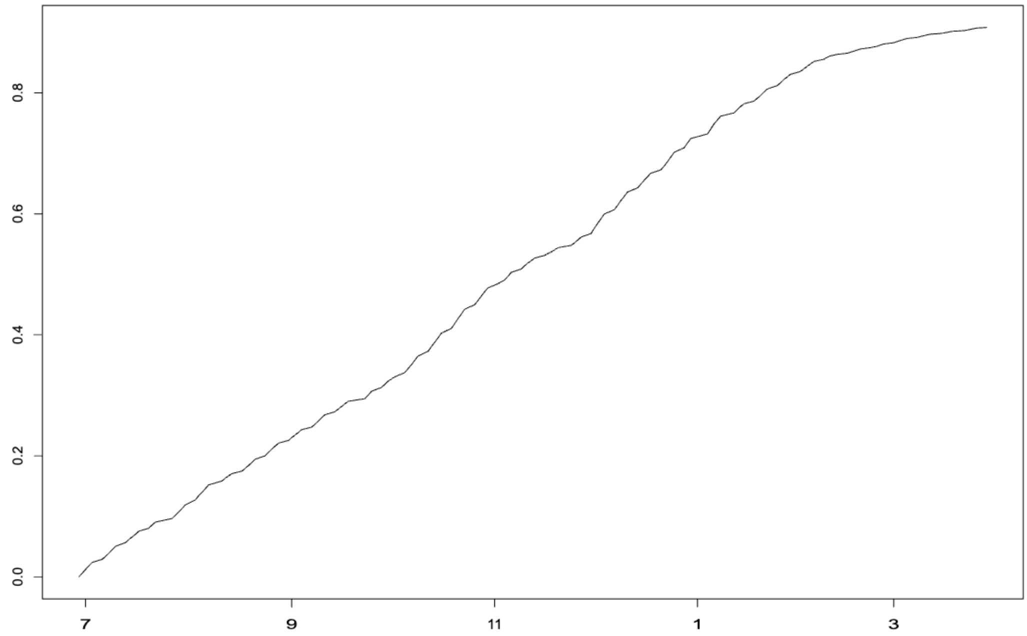

図 2.5　ゆうちょ銀行 2020 年度 状態 1 の確率

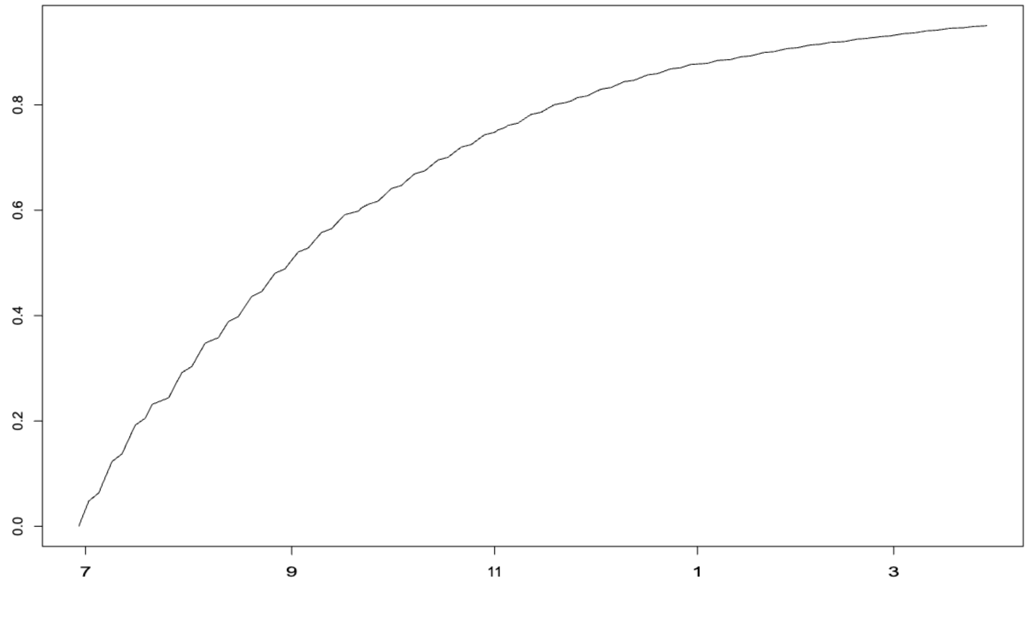

図 2.6　ゆうちょ銀行 2021 年度 状態 1 の確率

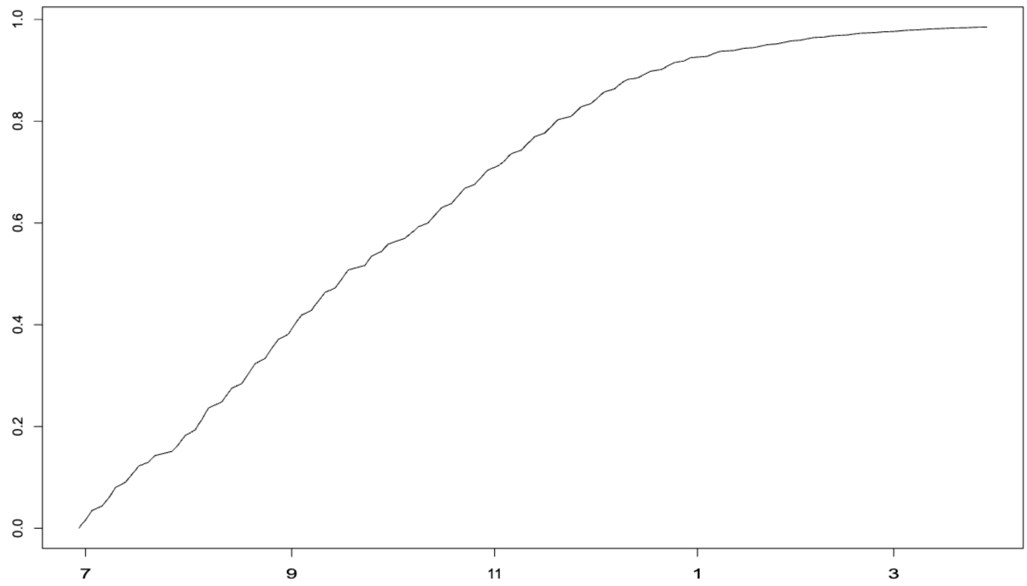

図 2.7　ハードオフ 2020 年度 状態 1 の確率

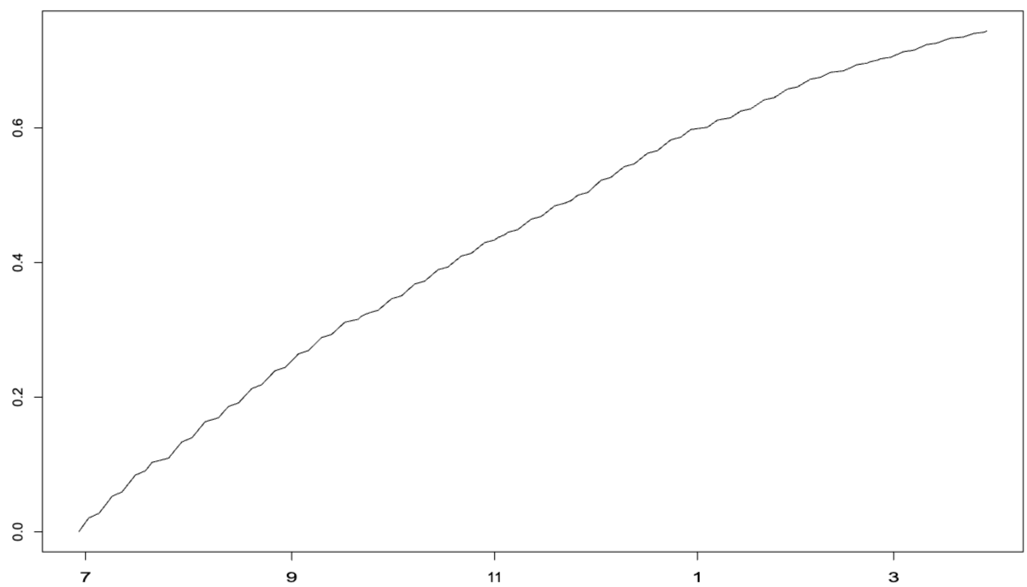

図 2.8　ハードオフ 2021 年度 状態 1 の確率

<table>
<tr><th>企業名</th><th>年度</th><th></th><th>σ</th><th>μ_0</th><th>μ_1</th></tr>
<tr><td rowspan="4">旭化成</td><td rowspan="2">上半期</td><td>推計値</td><td>0.2876533</td><td>0.009112037</td><td>0.1664644</td></tr>
<tr><td>標準誤差</td><td>5.77334E-05</td><td>0.053258051</td><td>0.068619217</td></tr>
<tr><td rowspan="2">下半期</td><td>推計値</td><td>0.34518032</td><td>-0.5490604</td><td>-0.02621885</td></tr>
<tr><td>標準誤差</td><td>7.88852E-05</td><td>0.057141191</td><td>0.109434745</td></tr>
<tr><td rowspan="4">日東工業</td><td rowspan="2">上半期</td><td>推計値</td><td>0.2594873</td><td>-0.5662943</td><td>0.433552</td></tr>
<tr><td>標準誤差</td><td>4.905E-05</td><td>0.044755376</td><td>0.071992971</td></tr>
<tr><td rowspan="2">下半期</td><td>推計値</td><td>0.2571181</td><td>-0.7897349</td><td>-0.2991414</td></tr>
<tr><td>標準誤差</td><td>4.66737E-05</td><td>0.041404542</td><td>0.091597495</td></tr>
</table>

<table>
<tr><th>企業名</th><th>年度</th><th></th><th>p_{01}</th><th>p_{10}</th></tr>
<tr><td rowspan="4">旭化成</td><td rowspan="2">上半期</td><td>推計値</td><td>2.4908E-13</td><td>0.02276982</td></tr>
<tr><td>標準誤差</td><td>6.0407E-09</td><td>0.002310221</td></tr>
<tr><td rowspan="2">下半期</td><td>推計値</td><td>9.98769E-11</td><td>0.04127302</td></tr>
<tr><td>標準誤差</td><td>1.11697E-07</td><td>0.004024475</td></tr>
<tr><td rowspan="4">日東工業</td><td rowspan="2">上半期</td><td>推計値</td><td>7.36E-17</td><td>0.03355923</td></tr>
<tr><td>標準誤差</td><td>1.20805E-10</td><td>0.003264148</td></tr>
<tr><td rowspan="2">下半期</td><td>推計値</td><td>7.31484E-13</td><td>0.05172063</td></tr>
<tr><td>標準誤差</td><td>9.31959E-09</td><td>0.005028315</td></tr>
</table>

表 2.4　推計結果 (年率). μ_i については r_i の標準誤差を用いている.

2.3.2　カルマンフィルター

続いて検証の第 2 段階であるカルマンフィルターの設定とデータを用いて分析, その結果について検討する. 市場が配当を強く意識し始めることで, 本来あるべき株価の動きから上方に逸脱していくことが第 1 段階において確認されていることから, 以下では状態 1 に遷移した時刻 τ から株価は異常な動きをするとみなす. したがって, カルマンフィルターは時刻 τ 以降の株価データに対して実行し, 本来あるべき理論株価のパス $\{\alpha_t\}$ を推計, データとして観測されている権利落ち日の始値と比較を行う.

ここでは以下のような線形システムを仮定する.

$$\begin{aligned} Y_t &= \alpha_t + \frac{t}{T}\{\log X_{T-} - \log(X_{T-} - D)\} + \sqrt{\Delta_t}\epsilon_t \\ \alpha_{t+1} &= \alpha_t + \delta\Delta_t + \sqrt{\Delta_t}\eta_t \end{aligned}$$

ここで X_{T-} は権利確定日の終値であり, Δ_t は各サンプルポイント間の時間幅, ϵ_t と η_t はそれぞれパラメータが $\sigma_\epsilon, \sigma_\eta$ である独立なホワイトノイズとする. $\{\alpha_t\}$ は本来の配当割引モデルにおける株価プロセスのパスであり, 状態 1 に移行したフェーズにおいてはデータとして不可視なものである. データとして観測されている $\{Y_t\}$ は時間経過によって α_t から $\{\log X_{T-} - \log(X_{T-} - D)\}$ の割合で乖離していき, 終端の時刻 T において予想される配当分 D の下落が発生すると考えている. 今株価は (2.1) に従う, すなわち幾何ブラウン運動であると仮定しているので, $\{\alpha_t\}$ はドリフト δ を持つ単位根過程としている.

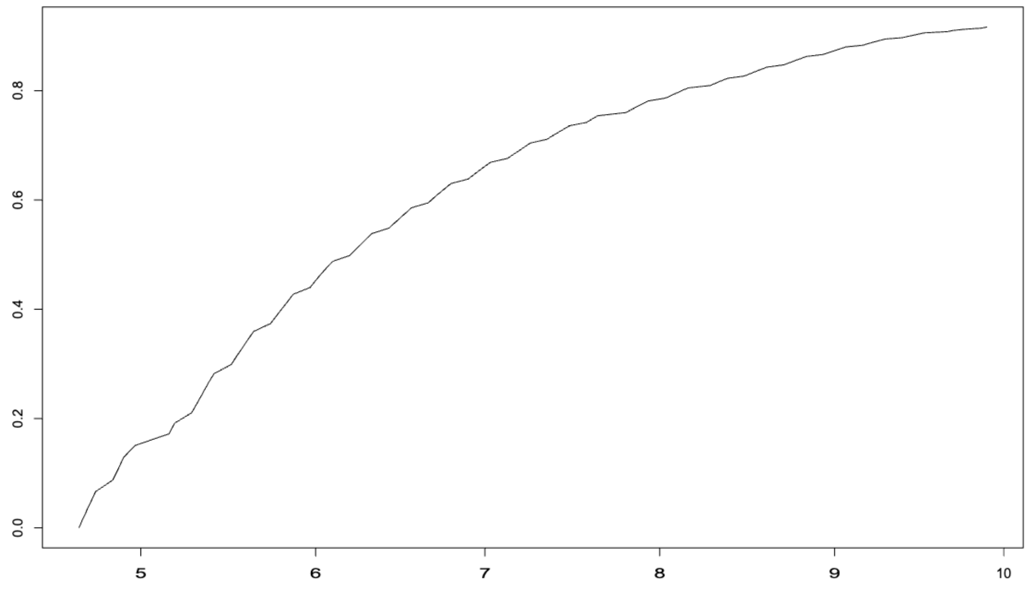

図 2.9　旭化成 2021 年度上半期 状態 1 の確率

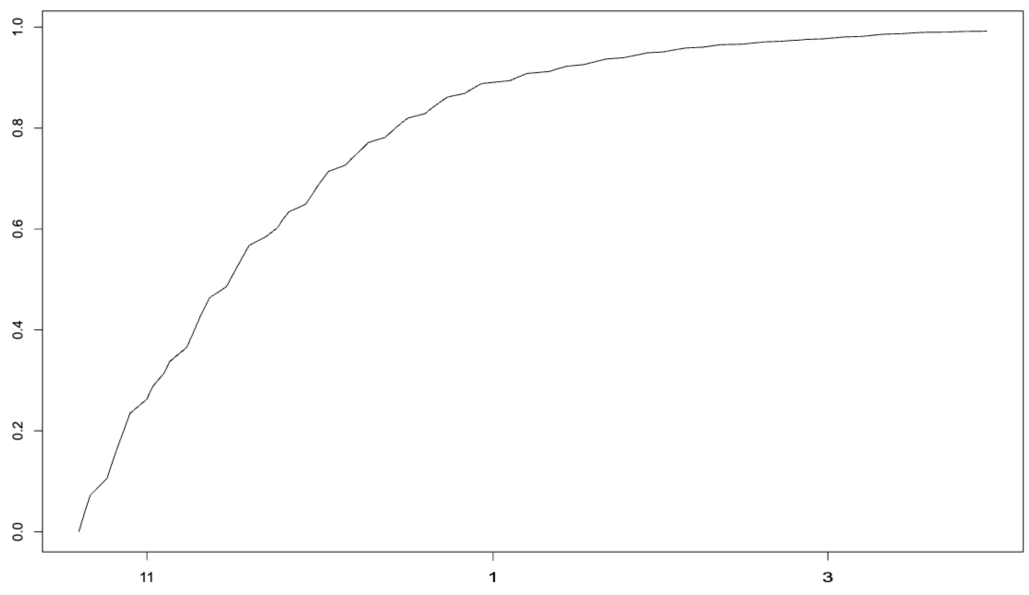

図 2.10　旭化成 2021 年度下半期 状態 1 の確率

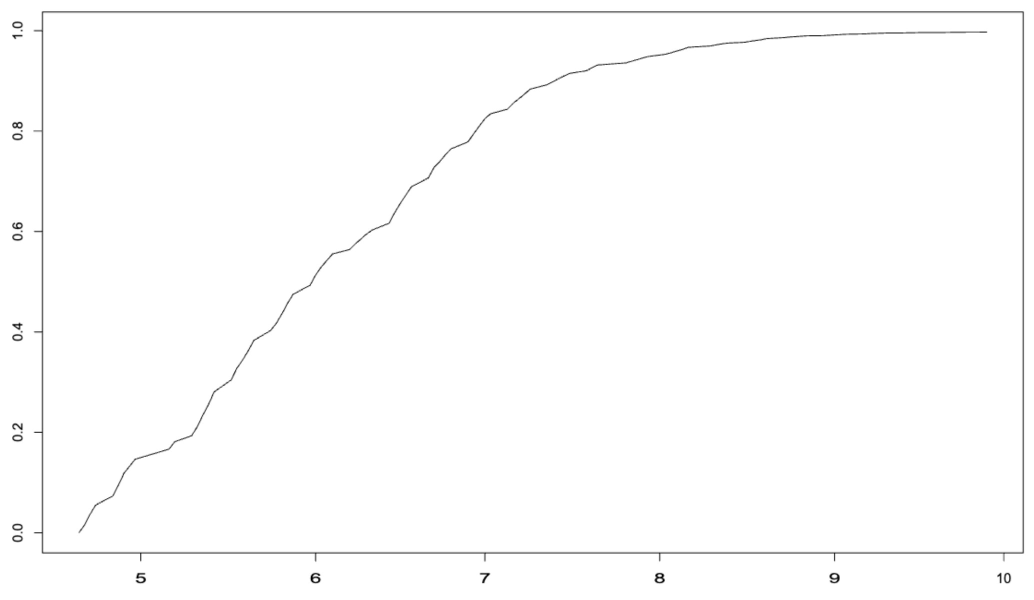

図 2.11　日東工業 2021 年度上半期 状態 1 の確率

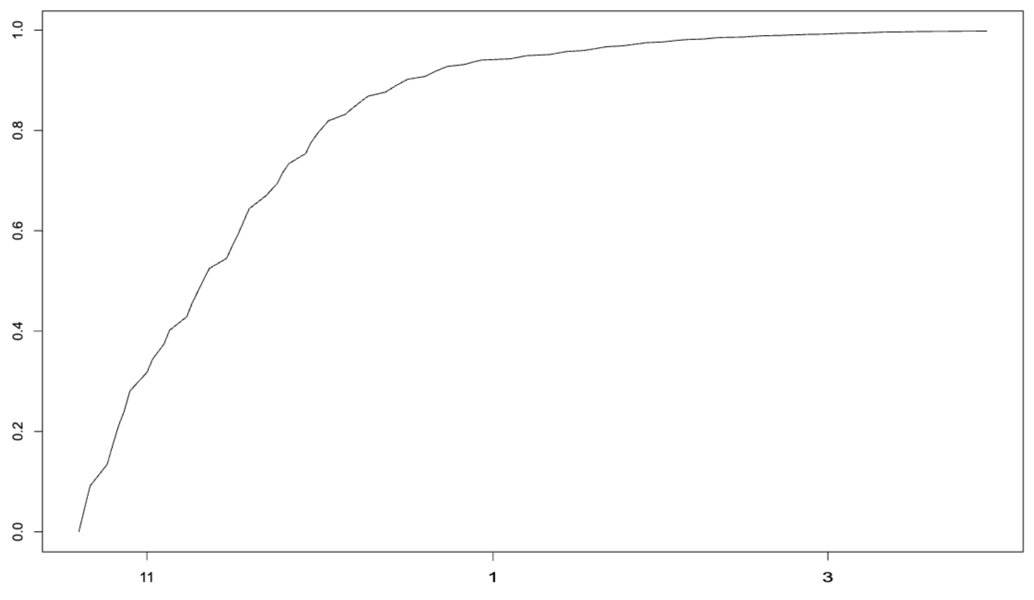

図 2.12　日東工業 2021 年度下半期 状態 1 の確率

上記の線形モデルは以下のように (z_t, β_t) の式として変換することでローカルトレンドモデル (cf. Tsay (2010) P. 558) の形に帰着できる.

$$\begin{cases} \beta_t &= \alpha_t - \delta t \\ z_t &= Y_t - \frac{t}{T}\{\log X_{T-} - \log(X_{T-} - D)\} - \delta t \end{cases}$$

なお, ローカルトレンドモデルにおけるカルマンフィルターについては付録 B を確認のこと.

以下では状態 1 に移行した時刻を $\tau = \min_t(\mathbb{P}(S_t = 1) > 0.6)$ とし, 時刻 τ からカルマンフィルターを適用することとし, 状態 1 に移行した時刻は配当を意識することによって起こる乖離が存在しないものとする. すなわちフィルタリングの初期値は z_τ である.

期末一括配当

期末一括配当銘柄について検証を行った結果は以下の表 2.5, 2.6 と図 2.13〜2.16 である. 表 2.5 はそれぞれのデータにおける $(\sigma_\epsilon, \sigma_\eta, \delta)$ の推計値, 表 2.6 は権利落ち日の始値とカルマンフィルターによって推計された α_T の値であり, 図は実際の株価のデータのパス $\{Y_t\}$ (実線) と推計された本来のモデルのパス $\{\alpha_t\}$ (破線) を表したものである.

表 2.6 にある結果のとおり, 推計したパスの終端は権利落ち日の始値の値と十分な精度で合致することが確認でき, 権利確定による株価の下落を捉えられている. 特筆すべき点は 2022 年 2 月 24 日に起こったロシアによるウクライナ侵攻によって全世界的に株価は暴落したが, その下落も α_t がしっかり捉えているところである. モデルの設定上, Y_t と α_t が比例配分による線形関係にあるため, こういった暴落もしっかりと捉えることが可能となっている.

しかし, ローカルトレンドモデルによって推定されたドリフト δ の値はレジームスイッチングモデルで推計された過熱時のドリフト $\mu_1(r_1)$ の値に近いことも確認できる. これは配当への期待が過剰であり, 配当額以上の上昇を市場が起こしている結果だと考えられ, このような場合, 直後の下落だけでなく数日間にわたって激しく株価が変動し, 最終的に平常時の株価で落ち着くことが多い. 今回は明確にデータとして存在する配当確定直後の株価 (始値) の推計を行っており, 権利落ち日以降の調整部分を考えていないため, 配当額以上の過剰な期待によってドリフトが大きく上に引っ張られていることは自然な結果である.

企業名	年度	σ_ϵ	σ_η	δ
ゆうちょ銀行	2020 年度	0.0001698048	0.1683386203	0.7836219535
	2021 年度	0.04512366	0.22335355	0.23964609
ハードオフ	2020 年度	0.1147628	0.1473191	0.5857408
	2021 年度	0.0855020	0.1521737	0.2302181

表 2.5　推計されたパラメータの値 (年率)

中間配当あり

中間配当がある銘柄について検証を行った結果は以下の表 2.7, 2.8 と図 2.17〜2.20 である. 表 2.7 はそれぞれのデータにおける $(\sigma_\epsilon, \sigma_\eta, \delta)$ の推計値, 表 2.8 は権利落ち日の始値とカルマンフィルターによって推計された α_T の値であり, 図は実際の株価のデータのパス $\{Y_t\}$ (実線) と推計された本来のモデルのパス $\{\alpha_t\}$ (破線) を表したものである.

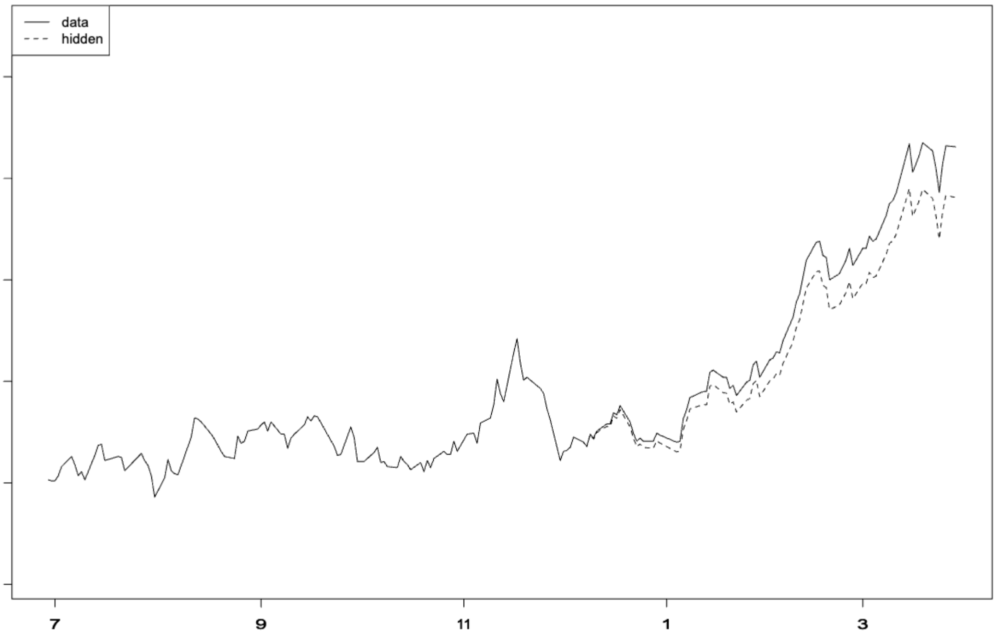

図 2.13　ゆうちょ銀行 2020 年度

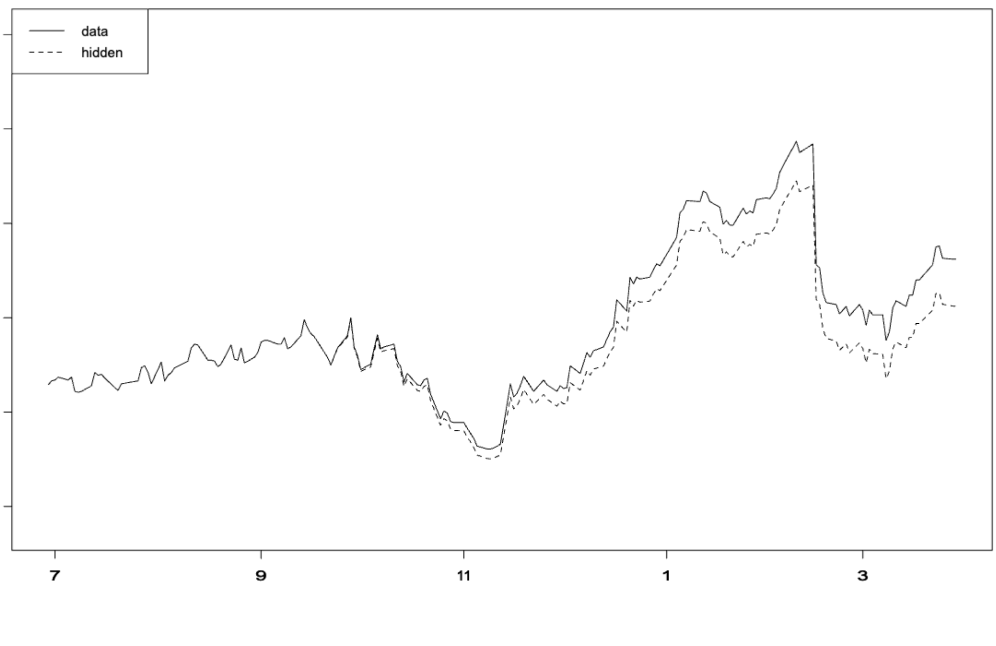

図 2.14　ゆうちょ銀行 2021 年度

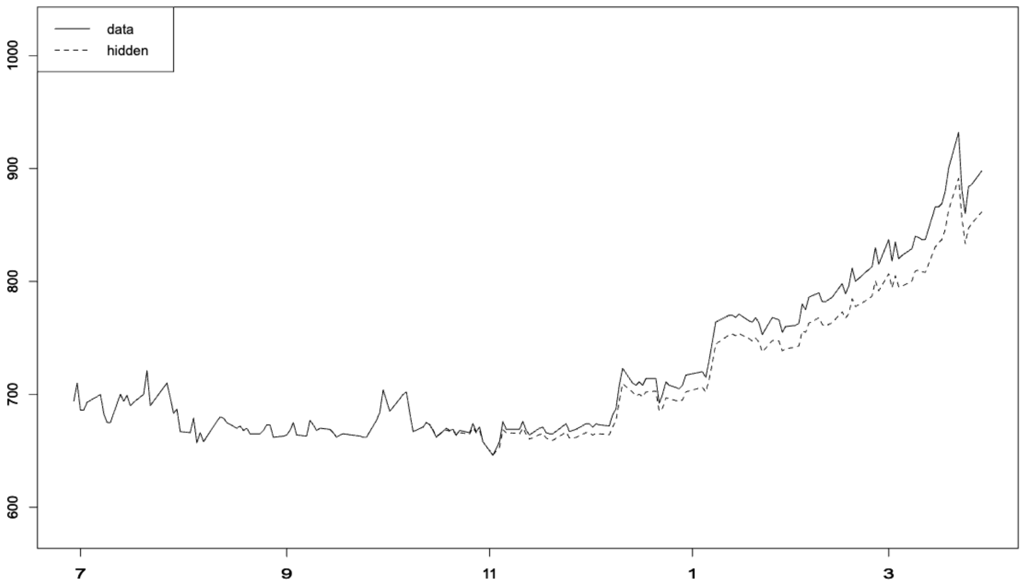

図 2.15　ハードオフ 2020 年度

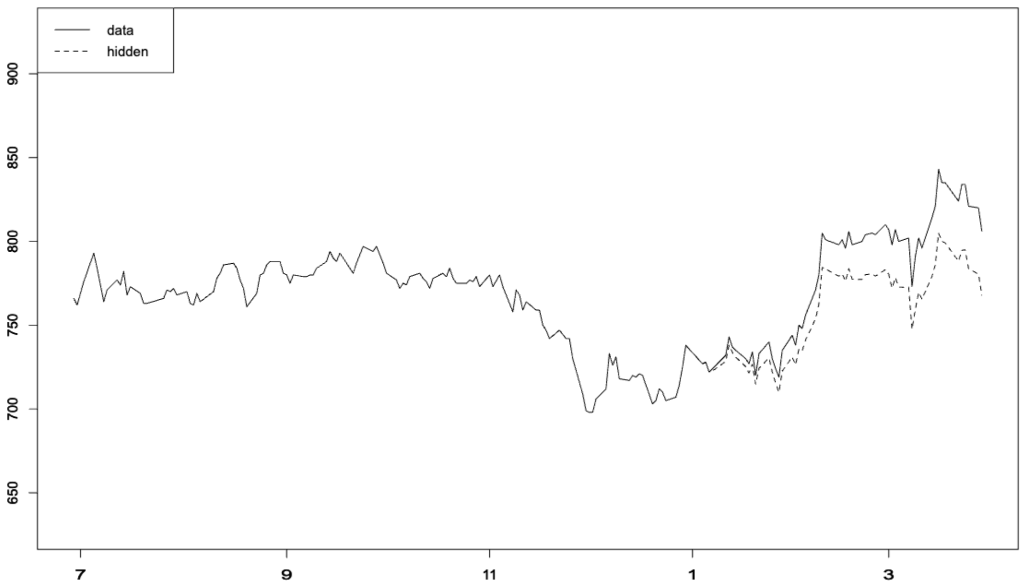

図 2.16　ハードオフ 2021 年度

企業名	年度	日付	始値	α_T
ゆうちょ銀行	2020 年度	3 月 30 日	1080	1081
	2021 年度	3 月 30 日	1010	1012.0197
ハードオフ	2020 年度	3 月 30 日	851	861.5245
	2021 年度	3 月 30 日	766	767.8029

表 2.6　配当確定後の株価のデータと推計された株価; 期末一括配当

表 2.8 にある結果を見ると, 期末一括配当銘柄よりも推計の精度は下がっている. これはレジームスイッチングモデルのパラメータ推定に用いたデータ数の差が 1 つの要因であると考えられる. また, 推定された α_t のパスであるが, 配当利回りは高いが, 年度中に 2 回に分けて支払われるため, 1 回 1 回の配当の大きさは小さくなり, 期末一括配当銘柄と比較すると乖離が小さくなるということが確認できる. また, ドリフト δ についても期末一括配当と同様の結果が確認される.

企業名	期間	σ_ϵ	σ_η	δ
旭化成	上半期	$3.760622 * 10^{-5}$	0.1945771	0.1920378
	下半期	0.0004412229	0.2584839058	-0.3430841328
日東工業	上半期	0.1662292	0.1266583	0.4426067
	下半期	0.0992967	0.1595216	-0.1780125

表 2.7　推計されたパラメータの値 (年率)

企業名	期間	日付	始値	α_T
旭化成	上半期	9 月 29 日	1224	1238.000
	下半期	3 月 30 日	1104	1097.0000
日東工業	上半期	9 月 29 日	2090	2119.468
	下半期	3 月 30 日	1611	1601.077

表 2.8　配当確定後の株価のデータと推計された株価; 中間配当あり

2.4　配当確定後株価の予測

ここまでは 1 年のデータを利用して当期配当による過熱が起こらない理論株価のパスを推計した. 最後に 3.1 節, 3.2 節の手法と株価のシミュレーションによって, 事前に市場参加者が権利付最終日後の価格を捕捉できるか検証を行う.

ここで σ はあらかじめ過去のデータから計算しているものとし, リアルタイムでドリフトの変化を捉え, そのドリフトを用いて将来の株価のパスをシミュレーションする. そしてシミュレーションしたパスからカルマンフィルターによって理論株価のパスを推計し, 配当確定後の価格を見る. 具体的な手順は以下の通りである.

1. t_n 時点までのデータを用いて, 3.1 節と同様に状態確率と (σ を除く) パラメータの推計を行う

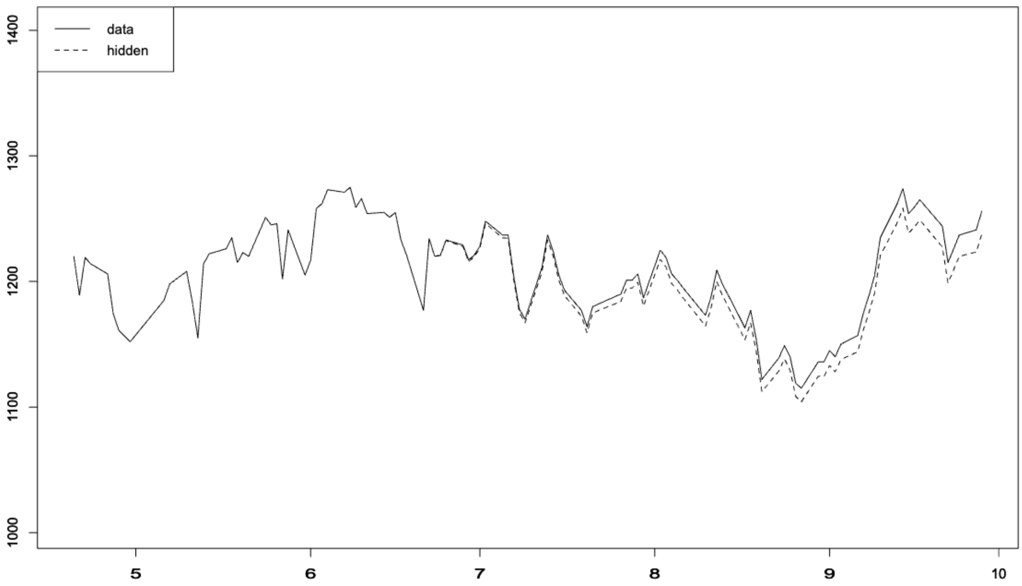

図 2.17　旭化成 2021 年度上半期

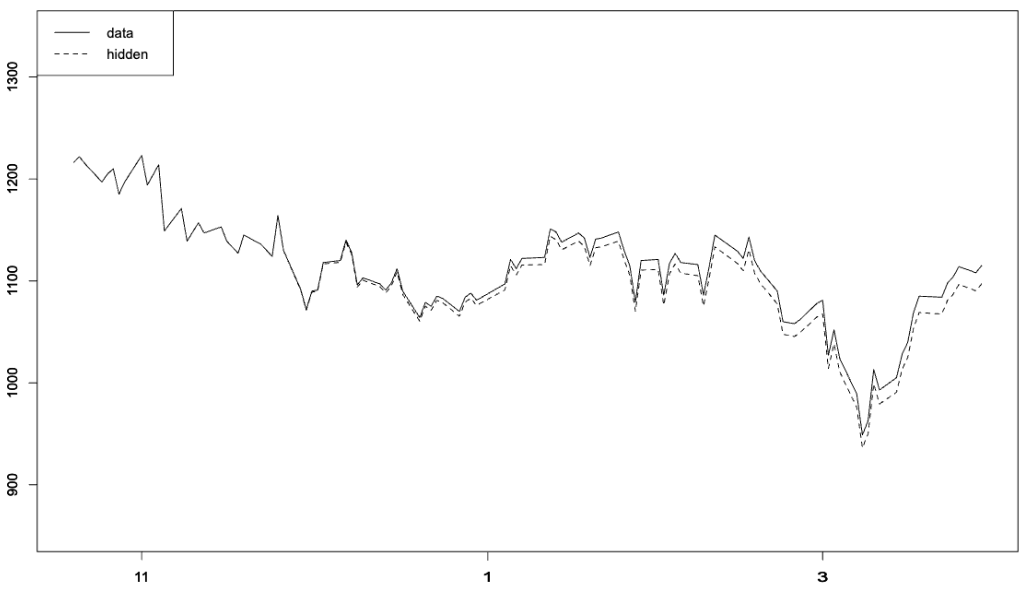

図 2.18　旭化成 2021 年度下半期

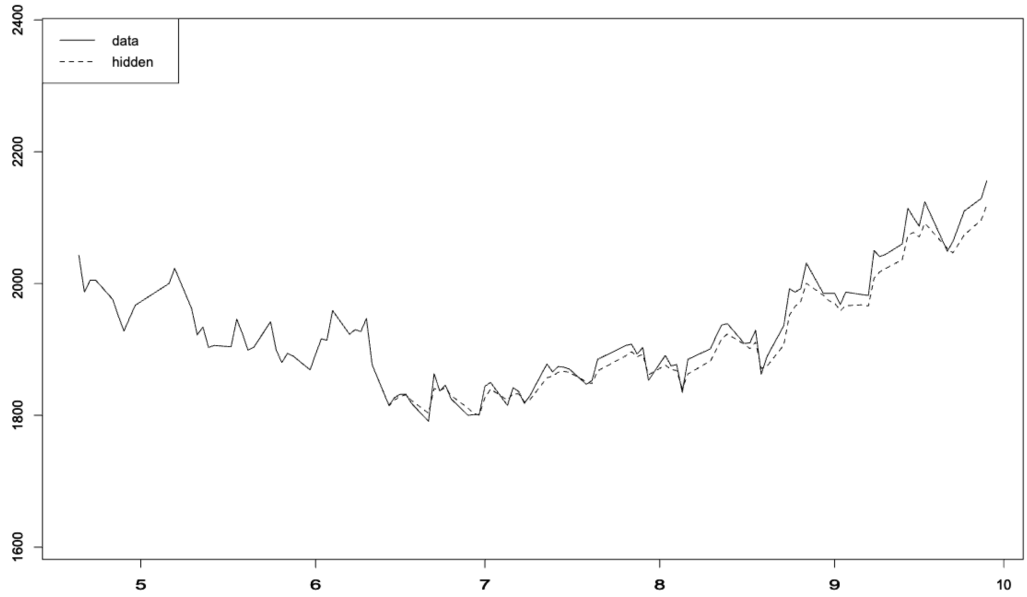

図 2.19　日東工業 2021 年度上半期

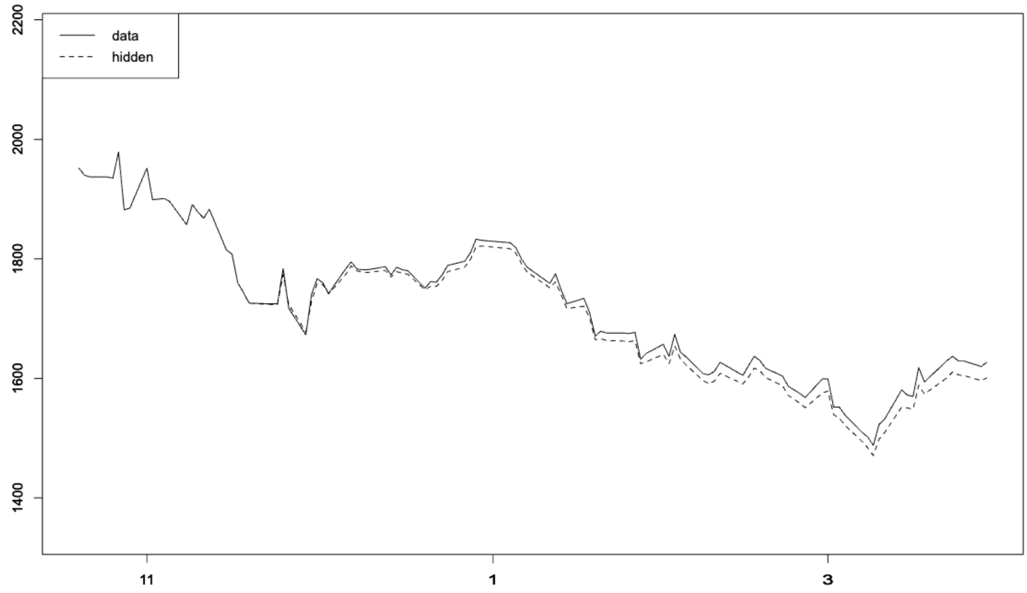

図 2.20　日東工業 2021 年度下半期

2. $\mu_0 < \mu_1$ であることを確認した上で, 状態 1 にいる状態確率が 0.6 を超えた点を τ と設定する
3. 仮にそのような τ が存在しない場合は $\tau = t_n$ とする
4. $(t_n, T]$ までの株価の対数系列のパスを 1 で推計した (μ_1, σ) によってシミュレーションする
5. データ $\{Y_t\}_{\tau \le t \le t_n}$ と 3 で作成したサンプルパスを用いて, 3.2 節と同様の処理を行う

なお, ここではゆうちょ銀行の 2020 年度のデータを用い, 11 月 16 日以降 10 営業日間隔で 2 月末まで代表点を取り, それぞれ推計を行った. 以下の表 2.9 はその結果である.

<table>
<tr><th>日付</th><th>推計値</th><th>標準誤差</th><th>実際の始値</th></tr>
<tr><td>2020 年 11 月 16 日</td><td>1072.4627</td><td>1.5113122</td><td rowspan="7">1080</td></tr>
<tr><td>2020 年 12 月 2 日</td><td>800.5312</td><td>1.0197476</td></tr>
<tr><td>2020 年 12 月 17 日</td><td>853.3827</td><td>0.9520759</td></tr>
<tr><td>2021 年 1 月 5 日</td><td>806.8956</td><td>0.8021015</td></tr>
<tr><td>2021 年 1 月 21 日</td><td>886.5152</td><td>0.8979321</td></tr>
<tr><td>2021 年 2 月 5 日</td><td>934.2802</td><td>0.7958022</td></tr>
<tr><td>2021 年 2 月 24 日</td><td>1013.3976</td><td>0.6676675</td></tr>
</table>

表 2.9　各時刻からの推計値と実際の始値

表の通り, 推計値は大きく外れてしまうことが多い. いくつか原因が考えられるが, 一番大きなものとしてカルマンフィルターを行う線形制御モデルの比例配分部が X_T に強く依存していることが考えられる. つまり X_T をシミュレーションによって正確に推計することができれば, 配当確定後の価格のより正確な推計につながると考えられる.

2.5　結論と今後の展望について

期末に向けて過熱し, 該当の期間が終了するとともに大幅に下落する株価を, 当期配当に対する過剰な意識によるものと仮定し, レジームスイッチングモデルとカルマンフィルターによって捉える手法を確立, その手法を用いて実際のデータが仮説を満たしているかを検証した. まずレジームスイッチングモデルによって特定の時期から株価のドリフトは大きくなる, つまり株式市場は期末に向けて過熱することが確認された. 市場が過熱しているという事実を踏まえ, カルマンフィルターによって配当確定後の株価を推計したところ, 期末終了日の翌日の始値に近い値をとり, この手法の有効性を得ることができた. しかし, ローカルトレンドモデルによって推定されたドリフト δ の値はレジームスイッチングモデルで推計された過熱時のドリフト $\mu_1(r_1)$ の値に近いことも確認された. これは配当への期待が過剰であり, 配当額以上の上昇を市場が起こしている結果だと考えられる. このような場合, 直後の下落だけでなく数日間にわたって激しく株価が変動し, 最終的に平常時の株価で落ち着くことが多い. 今回この論文では明確にデータとして存在する配当確定直後の株価の推計を行っており, 権利落ち日以降の調整部分を考えていないため, 配当額以上の過剰な期待によってドリフトが大きく上に引っ張られているのは自然な結果である. もし数日の激しい変動ののち, 安定した際の株価の推計を行う場合は $\delta = r_0$ とした上で, 追加で配当額以上の過剰な期待を表す変数を Y_t と α_t の関係式に追加して推計するといった方法が考えられる.

今後の展望としては 3 点考えられる. 1 つは線形のフィルタリング手法であるカルマンフィルターではない,

非線形フィルタリングへの応用が考えられる. 今回は単純化のために株価モデルには幾何ブラウン運動を用いて離散化, 加えて実現値と状態変数の関係を比例配分という線形関係で表した. このため線形フィルタリング手法であるカルマンフィルターを用いたが, データと状態変数の関係をより一般的にした場合は粒子フィルターのような非線形フィルタリングを扱うことになる. 比例配分とは異なる実現値とデータの関係を仮定した場合に精度が上がるかについて考察することは有意義であると考えられる.

2 点目は 3.3 節でも述べたとおり, 配当確定後の下落を捉えられるかは権利確定日の終値の推測精度に強く依存している. 権利確定日の終値の推測精度を上げることも今後の課題の 1 つであり, そのためにはクロスセクションデータを用いた当期配当によるドリフト上昇の傾向について検証・考察を行うことが考えられる.

3 点目は業績の変化による増配, あるいは減配の発表があった場合の株価の動きについてである. 増配があった場合, 将来キャッシュフロー自体が増加するため大きく株価が上昇する. この発表の時期が期末に近い場合, 配当による過熱がその上昇に吸収されることがある. この状況ではレジームスイッチングモデルで変化を捉えることが難しくなり, 今回のような分析が不可能になってしまう. このような場合の株価の短期モデルについても考察をすることは有意義であるため, 今後も取り組む予定である.

付録 2.A　レジームスイッチングモデルの概要

詳細は Kim, Nelson, et al. (1999) Chapter 4 を参照すること. この付録では分析の状況に合わせてレジームは $I = \{0, 1\}$ であるので $M = 2$, 情報集合を $\psi_t = \{\psi_{t-1}, Y_t\}$, 推移確率は時間によらず一定で p_{ij}, 設定より初期値を $\mathbb{P}(S_0 = 0|\psi_0) = 1$ とする.

2.A.1　フィルタリング

フィルター化確率 $p^f_{i,t} = \mathbb{P}(S_t = i|\psi_t)$ を求めることをフィルタリングと呼ぶ. t 期初において $p^f_{i,t-1}$ が所与であるとし, フィルタリングのアルゴリズムを確認する.

Step 1. $\mathbb{P}(S_t = j, S_{t-1} = i|\psi_{t-1})$ を以下のように求める.

$$\begin{aligned}\mathbb{P}(S_t = j, S_{t-1} = i|\psi_{t-1}) &= \mathbb{P}(S_t = j|S_{t-1} = i)\mathbb{P}(S_{t-1} = i|\psi_{t-1})\\ &= p_{ji} * p^f_{i,t-1}\end{aligned}$$

Step 2. t 時のデータを追加した場合の確率 $\mathbb{P}(S_t = j, S_{t-1} = i|\psi_t)$ を以下のように求める.

$$\begin{aligned}\mathbb{P}(S_t = j, S_{t-1} = i|\psi_t) &= \mathbb{P}(S_t = j, S_{t-1} = i|\psi_{t-1}, Y_t)\\ &= \frac{f(S_t = j, S_{t-1} = i, Y_t|\psi_{t-1})}{f(Y_t|\psi_{t-1})}\\ &= \frac{f(Y_t|S_t = j, S_{t-1} = i, \psi_{t-1})\mathbb{P}(S_t = j|S_{t-1} = i)\mathbb{P}(S_{t-1} = i|\psi_{t-1})}{\sum_j \sum_i f(Y_t|S_t = j, S_{t-1} = i, \psi_{t-1})\mathbb{P}(S_t = j|S_{t-1} = i)\mathbb{P}(S_{t-1} = i|\psi_{t-1})}\end{aligned}$$

この 2 つのステップにより $\mathbb{P}(S_t = j, S_{t-1} = i|\psi_t)$ が求まるので, S_{t-1} について総和をとれば $p^f_{i,t}$ が計算でき, 逐次的にフィルター化確率を求めることができる. なお $f(Y_t|S_t = j, S_{t-1} = i, \psi_{t-1})$ は平均 $r_j * \Delta_t$, 分散 $\sigma^2 * \sqrt{\Delta_t}$ の正規分布の密度関数の $Y_t - Y_{t-1}$ の値である.

2.A.2　平滑化 (Kim's Smoothing Algorithm)

利用可能な情報全てを用いて計算された各時点の状態確率である平滑化確率 $p^s_{i,t} = \mathbb{P}(S_t = i|\psi_T)$ を求めることを平滑化と呼ぶ. 以下ではフィルタリングによって $p^f_{i,T} = \mathbb{P}(S_T = i|\psi_T)$ が計算されているとし, それを初期値とする.

以下では $p^s_{i,t+1}$ を所与とし, 平滑化確率を求めるために $\mathbb{P}(S_t = j, S_{t+1} = k|\psi_T)$ を計算する.

$$\begin{aligned}\mathbb{P}(S_t = j, S_{t+1} = k|\psi_T) &= \mathbb{P}(S_{t+1} = k|\psi_T)\mathbb{P}(S_t = j|S_{t+1} = k, \psi_T)\\ &= \mathbb{P}(S_{t+1} = k|\psi_T)\mathbb{P}(S_t = j|S_{t+1} = k, \psi_t)\\ &= \frac{\mathbb{P}(S_{t+1} = k|\psi_T)\mathbb{P}(S_t = j, S_{t+1} = k, |\psi_t)}{\mathbb{P}(S_{t+1} = k|\psi_t)}\\ &= \frac{p^s_{k,t+1} p^f_{j,t} p_{kj}}{\sum_j p_{kj} * p^f_{j,t}}\end{aligned}$$

2 つ目の等号は $t+1$ 時以降の $Y.$ に対して S_t が S_{t+1} と ψ_t 以上の情報を持たないことから, $f((y_{t+1}, \cdots y_T)|S_{t+1} =$

$k, S_t = j, \psi_t) = f((y_{t+1}, \cdots y_T)|S_{t+1} = k, \psi_t)$ が成立する事実を利用して

$$
\begin{aligned}
\mathbb{P}(S_t = j|S_{t+1} = k, \psi_T) &= \frac{f(S_t = j, (y_{t+1}, \cdots y_T)|S_{t+1} = k, \psi_t)}{f((y_{t+1}, \cdots y_T)|S_{t+1} = k, \psi_t)} \\
&= \frac{f((y_{t+1}, \cdots y_T)|S_{t+1} = k, S_t = j\psi_t)\mathbb{P}(S_t = j|S_{t+1} = k, \psi_t)}{f((y_{t+1}, \cdots y_T)|S_{t+1} = k, \psi_t)} \\
&= \mathbb{P}(S_t = j|S_{t+1} = k, \psi_t)
\end{aligned}
$$

から成立する. 最後の等号はフィルタリングの Step.1 で計算した結果を用いている. これにより $p^f_{i,T}$ を初期値として逐次的に平滑化確率を計算することができる.

2.A.3 EM アルゴリズム

この小節では Kim, Nelson, et al. (1999) に加えて, 小西, 越智, 大森 (2008) を参考にした. EM アルゴリズムは特定したいモデルが欠測値, あるいは観測不可な変数を持っている場合にパラメータを推計するためのアルゴリズムである. 今回推計が必要なパラメータは $\boldsymbol{\theta} = (\sigma, r_0, r_1, p_{01}, p_{10})$ の 5 つであり, それぞれ $\boldsymbol{\theta}_0 = (\sigma, r_0, r_1), \boldsymbol{\theta}_1 = (p_{01}, p_{10})$ とする.

EM アルゴリズムは, パラメータ $\boldsymbol{\theta}^{(k)}$ と観測値によって計算される条件付き期待値を, 疑似的な観測値として欠測値に代入することで最大化する尤度関数を設定 (E ステップ) し, その尤度を最大化する $\boldsymbol{\theta}^{(k+1)}$ を求める (M ステップ)2 つのステップを収束するまで繰り返す推定手法である. ここでは欠測値は状態変数のベクトル $\mathbf{S}$ であることに注意して定式化すると, $\boldsymbol{\theta}^{(k)}$ を所与として

$$
\boldsymbol{\theta}^{(k+1)} = \arg\max_{\boldsymbol{\theta}\in\Theta} \mathbb{E}[l(\mathbf{Y}, \mathbb{E}[\mathbf{S}|\mathbf{Y}, \boldsymbol{\theta}^{(k)}]|\boldsymbol{\theta})|\mathbf{Y} = \mathbf{y}]
$$

を適当な初期値 $\boldsymbol{\theta}^{(0)}$ により, 収束するまでパラメータの更新を繰り返すアルゴリズムである. なお, $l(\cdot|\boldsymbol{\theta})$ は $\mathbf{Y}, \mathbf{S}$ が全て観測された場合の対数尤度関数であり, 最大化の目的関数は $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(k)})$ と表記する. 収束判定にはパラメータのノルム $\|\boldsymbol{\theta}^{(k+1)} - \boldsymbol{\theta}^{(k)}\|$ などが使用されるが, 今回はそれぞれパラメータの値が大きくないため, 尤度関数の差 $|Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(k+1)}) - Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(k)})|$ を収束判定に用いた.

$(\mathbf{Y}, \mathbf{S})$ の同時分布が $f(\mathbf{Y}, \mathbf{S}|\boldsymbol{\theta}) = f(\mathbf{Y}|\boldsymbol{S}, \boldsymbol{\theta}_1)f(\mathbf{S}|\boldsymbol{\theta}_2)$ であることを用いると, 目的の尤度関数は $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(k)})$ は

$$
\begin{aligned}
Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(k)}) &= \sum_{\mathbf{S}} f(\mathbf{Y}, \mathbf{S}|\boldsymbol{\theta}^{(k)}) \log f(\mathbf{Y}|\boldsymbol{S}, \boldsymbol{\theta}_1)f(\mathbf{S}|\boldsymbol{\theta}_2) \\
&= \sum_{\mathbf{S}} f(\mathbf{Y}, \mathbf{S}|\boldsymbol{\theta}^{(k)}) \log f(\mathbf{Y}|\boldsymbol{S}, \boldsymbol{\theta}_1) + \sum_{\mathbf{S}} f(\mathbf{Y}, \mathbf{S}|\boldsymbol{\theta}^{(k)}) \log f(\mathbf{S}|\boldsymbol{\theta}_2)
\end{aligned}
$$

と書ける. ここで $\sum_{\mathbf{S}}$ は各 $S_1, S_2, \cdots S_T$ に関する総和を意味する. これを用いてパラメータを推計する*2. 1 階条件をそれぞれのパラメータについて解くと

$$
\begin{aligned}
(\sigma^2)^{(k)} &= \sum_j \frac{\sum_t (Y_t - \Delta_t r_j^{(k+1)})^2 p_{j,t}^{s,(k)}}{\Delta_t \sum_t p_{j,t}^{s,(k)}} \\
r_j^{(k+1)} &= \frac{\sum_t \Delta_t Y_t p_{j,t}^{s,(k)}}{\sum_t \Delta_t^2 p_{j,t}^{s,(k)}} \\
p_{ji}^{(k+1)} &= \frac{\sum_t \mathbb{P}(S_t = j, S_{t-1} = i|\mathbf{Y}, \boldsymbol{\theta}^{(k)})}{\sum_t p_{i,t-1}^{s,(k)}}
\end{aligned}
$$

*2 変形など詳しい議論は Kim, Nelson, et al. (1999) を参照

である. また, 標準偏差は James D. Hamilton (1990) の議論等を元にヘッセ行列 $H(\boldsymbol{\theta})$ が

$$
\begin{aligned}
H(\boldsymbol{\theta}) &= \frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(k)})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} = \begin{pmatrix} \mathbf{P_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{P_2} \end{pmatrix} \\
\text{where} \quad \mathbf{P_1} &= \begin{pmatrix} \sum_t \sum_j \left(\frac{1}{2\sigma^4} - \frac{(Y_t - \Delta_t r_j)^2}{\sigma^6} \right) p^s_{j,t} & \sum_t -\frac{\Delta_t (Y_t - \Delta_t r_0)}{\sigma^4} p^s_{0,t} & \sum_t -\frac{\Delta_t (Y_t - \Delta_t r_1)}{\sigma^4} p^s_{1,t} \\ \sum_t -\frac{\Delta_t (Y_t - \Delta_t r_0)}{\sigma^4} p^s_{0,t} & \sum_t -\frac{\Delta_t^2}{\sigma^2} p^s_{0,t} & 0 \\ \sum_t -\frac{\Delta_t (Y_t - \Delta_t r_1)}{\sigma^4} p^s_{1,t} & 0 & \sum_t -\frac{\Delta_t^2}{\sigma^2} p^s_{0,t} \end{pmatrix} \\
\mathbf{P_2} &= \begin{pmatrix} \sum_t -\frac{1}{p_{01}^2} \mathbb{P}(S_t = 1, S_{t-1} = 0 | \psi_T) & 0 \\ 0 & \sum_t -\frac{1}{p_{10}^2} \mathbb{P}(S_t = 0, S_{t-1} = 1 | \psi_T) \end{pmatrix}
\end{aligned}
$$

と計算されることより, $V = -H(\hat{\boldsymbol{\theta}})^{-1}$ の対角要素の平方根として計算される.

付録 2.B　ローカルトレンドモデルに関して

以下では Tsay (2010) の内容を要約する.

2.B.1　カルマンフィルターの概要

詳細は Tsay (2010) PP.558-563 を参照のこと.

ローカルトレンドモデルは状態空間モデルの中で最も単純な線形モデルであり, 以下のような形のもののことを指す.

$$
\begin{aligned}
y_t &= \mu_t + e_t, & e_t &\sim N(0, \sigma_e^2) \\
\mu_{t+1} &= \mu_t + \eta_t, & \eta_t &\sim N(0, \sigma_\eta^2)
\end{aligned} \tag{2.2}
$$

ここで $\{y_t | t = 0, 1, \cdots, T\}$ は観測可能なデータ, μ_t は観測不可で初期値 μ_0 の値は所与, あるいは特定の既知の分布に従うものである. また, ホワイトノイズ e_t, η_t は互いに独立である. 状態空間モデルを使った分析にはフィルタリング, 予測, 平滑化の 3 つの形があるが, 今回はフィルタリングによってパラメータの推計や状態変数の分析を行う.

いくつかの記号についての説明を行う. μ_t の分散は Σ_t とし, 時刻 $\mathcal{F}_s$ で条件付けた確率変数の条件付き期待値を $y_{t|s}$ といった形で表現する. また 1 ステップ先の予測誤差は v_t, その分散は V_t とする. なお予測誤差 v_t は $\mathcal{F}_{t-1}$ と独立であるので, v_t の条件付き分散 $V_{t|t-1}$ は通常の分散と一致する. 以上をまとめると,

$$
\Sigma = \mathrm{Var}(\mu_t), \quad y_{t|t-1} = \mathbb{E}[y_t | \mathcal{F}_{t-1}], \quad v_t = y_t - y_{t|t-1}, \quad V_t = \mathrm{Var}(v_t | \mathcal{F}_{t-1})
$$

となる. (2.2) の式より

$$
y_{t|t-1} = \mathbb{E}[y_t | \mathcal{F}_{t-1}] = \mathbb{E}[\mu_t + e_t | \mathcal{F}_{t-1}] = \mu_{t|t-1}
$$

であるので, 予測誤差について以下の関係が成立する.

$$
\begin{aligned}
v_t &= y_t - y_{t|t-1} = y_t - \mu_{t|t-1} \\
V_t &= \mathrm{Var}(y_t - \mu_{t|t-1} | \mathcal{F}_{t-1}) = \mathrm{Var}(\mu_t + e_t - \mu_{t|t-1} | \mathcal{F}_{t-1}) \\
&= \mathrm{Var}(\mu_t - \mu_{t|t-1} | \mathcal{F}_{t-1}) + \mathrm{Var}(e_t | \mathcal{F}_{t-1}) \\
&= \Sigma_{t|t-1} + \sigma_e^2
\end{aligned}
$$

また, 簡単な計算から $\mathbb{E}[v_t] = 0$ と $\mathrm{Cov}(v_t, y_j) = 0, j < t$ がわかる. これより過去の y_j と 1 ステップ先の予測誤差には相関がなく, $\mu_{t|t} = \mathbb{E}[\mu_t|\mathcal{F}_t] = \mathbb{E}[\mu_t|\mathcal{F}_{t-1}, v_t]$ というように, $\mathcal{F}_t = \{\mathcal{F}_{t-1}, y_t\} = \{\mathcal{F}_{t-1}, v_t\}$ と情報集合を書き直すことが可能となる. $\mathcal{F}_{t-1}$ が所与の場合の (μ_t, v_t) の条件付き同時分布が

$$\begin{bmatrix}\mu_t \\ v_t\end{bmatrix}_{\mathcal{F}_{t-1}} = N\left(\begin{bmatrix}\mu_{t|t-1} \\ 0\end{bmatrix}, \begin{bmatrix}\Sigma_{t|t-1} & \Sigma_{t|t-1} \\ \Sigma_{t|t-1} & V_{t|t-1}\end{bmatrix}\right)$$

であることと, Tsay (2010) Theorem 11.1(多変量正規分布の性質) より

$$\begin{aligned}
K_t &:= \frac{\Sigma_{t|t-1}}{V_t} \qquad \text{(Kalman Gain)} \\
\mu_{t|t} &= \mu_{t|t-1} + \frac{\Sigma_{t|t-1} v_t}{V_t} = \mu_{t|t-1} + K_t v_t \\
\Sigma_{t|t} &= \Sigma_{t|t-1} - \frac{\Sigma_{t|t-1}^2}{V_t} = \Sigma_{t|t-1}(1 - K_t)
\end{aligned}$$

と求められる. この結果を (2.2) に適用することで $(\mu_{t+1|t}, \Sigma_{t+1|t})$ を計算することができ, これらを繰り返すことでカルマンフィルターのアルゴリズムが得られる. まとめると以下の通りである.

$$\begin{aligned}
v_t &= y_t - \mu_{t|t-1} \\
V_t &= \Sigma_{t|t-1} + \sigma_e^2 \\
K_t &= \frac{\Sigma_{t|t-1}}{V_t} \\
\mu_{t+1|t} &= \mu_{t|t-1} + K_t v_t \\
\Sigma_{t+1|t} &= \Sigma_{t|t-1}(1 - K_t) + \sigma_\eta^2
\end{aligned}$$

なお Tsay (2010) では初期分布 $N(\mu_{1|0}, \Sigma_{1|0})$ について考慮しているが, 今回は出発点 μ_0 の値が y_0 と確定しているため, $\Sigma_0 = 1000$ として分析を行なっている.

2.B.2 パラメータの推計に関して

詳細は Tsay (2010) PP.564-565 を参照のこと.

上記ローカルトレンドモデルは推計の必要なパラメータが (σ_e, σ_η) の 2 つである. これらを予測誤差の分布関数からなる対数尤度を用いた最尤法で求める. 1 ステップ先の予測誤差は以下のようにベクトル形式で書くことができる.

$$\mathbf{v} = \mathbf{K}(\mathbf{y} - \mu_{1|0}\mathbf{1}_T) \tag{2.3}$$

ただし, $\mathbf{v} = (v_1, \cdots, v_T)^\top, \mathbf{y} = (y_1, \cdots, y_T)^\top, \mathbf{1}_T$ は T 次ベクトルで全ての要素が 1 であるものとし, $\mathbf{K}$ は以下のような下三角行列として定義される.

$$\mathbf{K} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ k_{21} & 1 & 0 & \cdots & 0 \\ k_{31} & k_{32} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ k_{T1} & k_{T2} & k_{T3} & \cdots & 1 \end{bmatrix} \qquad \text{where} \qquad \begin{aligned} & k_{i,i-1} = -K_{i-1} \\ & k_{i,j} = -(1 - K_{i-1})(1 - K_{i-2}) \cdots (1 - K_{j+1})K_j \\ & (i = 1, 2, \cdots, T, j = 1, \cdots i - 2) \end{aligned}$$

(2.3) より, y_t から v_t への変換はヤコビ行列式が 1 になるため, $\mathbf{y}$ と $\mathbf{v}$ の同時密度関数は $p(\mathbf{y}) = p(\mathbf{v})$ の関係にある. これを用いれば

$$p(\mathbf{v}) = p(\mathbf{y}) = p(y_1)\prod_{j=2}^{T} p(y_j|\mathcal{F}_{j-1}) = p(v_1)\prod_{j=2}^{T} p(v_j) = \prod_{j=1}^{T} p(v_j)$$

である. $v_t \sim N(0, V_t)$ であることと各 (σ_e, σ_η) の組から (v_t, V_t) が計算されることを用いれば, 対数尤度関数は以下の通りである.

$$l(v_t|\sigma_e, \sigma_\eta) = \sum_{t=1}^{T}\left(-\log V_t - \frac{v_t^2}{2V_t^2}\right) + (const)$$

これを最大化する (σ_e, σ_η) を推計値として与える. なお, 今回 3.2 節で使用したモデルはローカルトレンドモデルへの変形をおこなったことにより, 実際は δ もパラメータに含まれているが, 最大化する対数尤度関数自体は同じである.

付録 2.C　Sample Codes

2.C.1　Data Analysis

Listing 2.1　Filtering by R 4.0.3

```
# index_202q <- c(1222, 1344)
# index_204q <- c(1344, 1466)
# index_212q <- c(1466, 1588)
# index_214q <- c(1588, 1710)

# loading libraries
library(xts)

# extract data from csv as list
# input csv includes ex-rights day
extract_data <- function(input_path, index = c(1, 0)) {
    data <- read.csv(input_path)
    n <- index[2] - 1
    date <- as.Date(data[index[1]:n, 1])
    duration <- as.numeric(date[length(date)] -  date)
    price <- ts(data[index[1]:n, 7])
    terminal_price <- data[n + 1, 2]
    quote <- ts(data[index[2]:n, 6])
    return(list(date, duration, price, terminal_price, quote))
}

# file paths
hardoff <- "csv/hardoff.csv"
yucho <- "csv/yucho.csv"
nitto <- "csv/nitto.csv"
asahikasei <- "csv/asahikasei.csv"

# data indices (exclude yucho)
index_1520 <- c(1, 1222)
index_2021 <- c(1283, 1466) # ex-rights day of 2019 4q index is 1222
index_2122 <- c(1527, 1710) # ex-rights day of 2020 4q index is 1466

# of half year
index_202q <- c(1237, 1344)
index_204q <- c(1359, 1466)
index_212q <- c(1481, 1588)
index_214q <- c(1603, 1710)

# data indices of yucho
index_yucho_1520 <- c(1, 1074)
index_yucho_2021 <- c(1135, 1318) # ex-rights day of 2019 4q index is 1074
index_yucho_2122 <- c(1379, 1562) # ex-rights day of 2020 4q index is 1318
```

```
# dividends
div_hardoff <- c(35, 40)
div_yucho <- c(50, 50)
div_nitto <- c(25, 25)
div_asahikasei <- c(18, 18)


# extracting data
yucho_1520 <- extract_data(yucho, index_yucho_1520)
yucho_2021 <- extract_data(yucho, index_yucho_2021)
yucho_2122 <- extract_data(yucho, index_yucho_2122)

hardoff_1520 <- extract_data(hardoff, index_1520)
hardoff_2021 <- extract_data(hardoff, index_2021)
hardoff_2122 <- extract_data(hardoff, index_2122)

nitto_1520 <- extract_data(nitto, index_1520)
nitto_202q <- extract_data(nitto, index_202q)
nitto_204q <- extract_data(nitto, index_204q)
nitto_212q <- extract_data(nitto, index_212q)
nitto_214q <- extract_data(nitto, index_214q)

asahikasei_1520 <- extract_data(asahikasei, index_1520)
asahikasei_202q <- extract_data(asahikasei, index_202q)
asahikasei_204q <- extract_data(asahikasei, index_204q)
asahikasei_212q <- extract_data(asahikasei, index_212q)
asahikasei_214q <- extract_data(asahikasei, index_214q)



# 1st step: regime switching model
# define functions
regime_switch <- function(par_dbm, par_bb, p, q, dt, price, duration) {
    n <- length(price)
    trans_prob <- matrix(c(1 - p, p, q, 1 - q), ncol = 2)
    r1 <- par_dbm[2] - par_dbm[1]^2 / 2
    r2 <- par_bb[2] - par_dbm[1]^2 / 2
    duration <- duration * dt
    diff_duration <- -diff(duration)
    log_price <- log(price)
    steady_prob <- matrix(numeric(2 * n), nrow = 2)
    filt_prob <- matrix(numeric(2 * (n - 1)), nrow = 2)

    # filtering step
    steady_prob[,1] <- c(0,1)
    likelihood <- 0
    trans_dens_dbm <- dnorm(diff(log_price), r1 * diff_duration ,par_dbm[1] * sqrt(diff_duration))
    trans_dens_bb <- dnorm(diff(log_price), r2 * diff_duration, par_dbm[1] * sqrt(diff_duration))
    for (i in 1:(n - 1)) {
        filt_prob[, i] <- trans_prob %*% steady_prob[, i]
        p_likelihood <- filt_prob[, i] * c(trans_dens_dbm[i], trans_dens_bb[i])
        steady_prob[, i + 1] <- p_likelihood / sum(p_likelihood)
        likelihood <- likelihood + log(sum(p_likelihood))
    }

    # smoothing step
    inv_filt <- 1 / filt_prob
    inv_filt[is.infinite(inv_filt)] <- 0
    smooth_prob <- matrix(numeric(2 * n), nrow = 2)
    smooth_prob[, n] <- steady_prob[, n]
    for (i in (n - 1):1){
        smooth_prob[1, i] <- inv_filt[1, i] * smooth_prob[1, i+1] * steady_prob[1, i] * (1 - p) + inv_filt
          [2, i] * smooth_prob[2, i + 1] * steady_prob[1, i] * p
        smooth_prob[2, i] <- inv_filt[2, i] * smooth_prob[2, i+1] * steady_prob[2, i] * (1 - q) + inv_filt
          [1, i] * smooth_prob[1, i + 1] * steady_prob[2, i] * q
        p_likelihood <- smooth_prob[, i] * c(trans_dens_dbm[i], trans_dens_bb[i])
        likelihood <- likelihood + log(sum(p_likelihood))
    }
    # EM algorithm
    r1_est <- sum(diff(log_price) * smooth_prob[1, -1]) / sum(diff_duration * smooth_prob[1, -1])
    r2_est <- sum(diff(log_price) * smooth_prob[2, -1]) / sum(diff_duration * smooth_prob[2, -1])
    # convert sigma yearly rate
    sig_est <- sqrt(1 / (n-1) * sum((diff(log_price) - diff_duration * r1_est)^2 / diff_duration * smooth_
      prob[1, -1] + (diff(log_price)- diff_duration * r2_est)^2 / diff_duration * smooth_prob[2, -1]))
    sig_dt <- sig_est * sqrt(diff_duration) # daily rate
    mu1_est <- r1_est + sig_est^2 / 2
    mu2_est <- r2_est + sig_est^2 / 2
    p_est <- sum(smooth_prob[2, -1] * p * steady_prob[1,-n] * inv_filt[2,]) / sum(smooth_prob[1, -n])
```

```
    q_est <- sum(smooth_prob[1, -1] * q * steady_prob[2,-n] * inv_filt[1,]) / sum(smooth_prob[2, -n])

    # calculate cov-matrix
    hessian_11 <- sum((1/(2*sig_dt^4)- (diff(log_price) - diff_duration * r1_est)^2/sig_dt^6) * smooth_prob[1, -1] + (1/(2*sig_dt^4)- (diff(log_price) - diff_duration * r2_est)^2/sig_dt^6) * smooth_prob[2, -1])
    hessian_12 <- sum(- diff_duration * (diff(log_price) - diff_duration * r1_est)/sig_dt^4 * smooth_prob[1, -1])
    hessian_13 <- sum(- diff_duration * (diff(log_price) - diff_duration * r2_est)/sig_dt^4 * smooth_prob[2, -1])
    hessian_22 <- sum(-diff_duration^2/sig_dt^2 * smooth_prob[1, -1])
    hessian_33 <- sum(-diff_duration^2/sig_dt^2 * smooth_prob[2, -1])
    hessian_p <- sum(- inv_filt[2, ] * smooth_prob[2, -1] * steady_prob[1, -n])/ p_est
    hessian_q <- sum(- inv_filt[1, ] * smooth_prob[1, -1] * steady_prob[2, -n])/ q_est
    hessian <- diag(c(hessian_11, hessian_22, hessian_33, hessian_p, hessian_q))
    hessian[2,1] <- hessian_12
    hessian[1,2] <- hessian_12
    hessian[3,1] <- hessian_13
    hessian[1,3] <- hessian_13
    variance <- sqrt(diag(-solve(hessian, tol=1e-50)))
    variance[1] <- variance[1] / sqrt(dt) # convert into yearly rate
    se <- variance / sqrt(n-1)
    return(list(steady_prob, filt_prob, smooth_prob, likelihood, c(sig_est, mu1_est, mu2_est, p_est, q_est), se))
}

em_function <- function(current_data, dt, trans_prob, q, m, epsilon, init_par1, init_par2){
    par_current_state1 <- init_par1
    par_current_state2 <- init_par2
    repeat_index <- 0
    current_ll <- 2 * epsilon
    archive_ll <- 0
    # while loop
    while (abs(current_ll - archive_ll) > epsilon && repeat_index < m) {
        archive_ll <- current_ll
        repeat_index <- repeat_index + 1
        list_regime <- regime_switch(par_current_state1, par_current_state2, trans_prob, q, dt, current_data[[3]], current_data[[2]])
        current_ll <- list_regime[[4]]
        par_current_state1 <- c(list_regime[[5]][1], list_regime[[5]][2])
        par_current_state2 <- c(list_regime[[5]][1], list_regime[[5]][3])
        trans_prob <- list_regime[[5]][4]
        q <- list_regime[[5]][5]
    }
    likelihood <- list_regime[[4]]
    se <- list_regime[[6]]
    smooth_prob_1 <- list_regime[[3]][1,]
    return(list(par_current_state1, par_current_state2, likelihood, repeat_index, current_data[[4]], smooth_prob_1, se, trans_prob, q))
}

dt <- 1 / 365
p_12 <- 10^-3
p_21 <- 10^-3
m <- 10000
epsilon <- 10^-8
init_par1 <- c(0.25, 1)
init_par2 <- c(0.25, 0.5)

# yucho
regime_yucho_2021 <- em_function(yucho_2021, dt, p_12, p_21, m, epsilon, init_par1, init_par2)
regime_yucho_2122 <- em_function(yucho_2122, dt, p_12, p_21, m, epsilon, init_par1, init_par2)

# hardoff
regime_hardoff_2021 <- em_function(hardoff_2021, dt, p_12, p_21, m, epsilon, init_par1, init_par2)
regime_hardoff_2122 <- em_function(hardoff_2122, dt, p_12, p_21, m, epsilon, init_par1, init_par2)

# nitto
regime_nitto_212q <- em_function(nitto_212q, dt, p_12, p_21, m, epsilon, init_par1, init_par2)
regime_nitto_214q <- em_function(nitto_214q, dt, p_12, p_21, m, epsilon, init_par1, init_par2)

# asahikasei
regime_asahikasei_212q <- em_function(asahikasei_212q, dt, p_12, p_21, m, epsilon, init_par1, init_par2)
regime_asahikasei_214q <- em_function(asahikasei_214q, dt, p_12, p_21, m, epsilon, init_par1, init_par2)

# plot smoothed probability
plot(yucho_2021[[1]], regime_yucho_2021[[6]], type="l", ann="F")
plot(yucho_2122[[1]], regime_yucho_2122[[6]], type="l", ann="F")
```

```
plot(hardoff_2021[[1]], regime_hardoff_2021[[6]], type="l", ann="F")
plot(hardoff_2122[[1]], regime_hardoff_2122[[6]], type="l", ann="F")

plot(nitto_212q[[1]], regime_nitto_212q[[6]], type="l", ann="F")
plot(nitto_214q[[1]], regime_nitto_214q[[6]], type="l", ann="F")

plot(asahikasei_212q[[1]], regime_asahikasei_212q[[6]], type="l", ann="F")
plot(asahikasei_214q[[1]], regime_asahikasei_214q[[6]], type="l", ann="F")


# check state index
prob_threshold <- 0.6
cpt_y0 <- min(which(regime_yucho_2021[[6]] > prob_threshold))
cpt_y1 <- min(which(regime_yucho_2122[[6]] > prob_threshold))

cpt_h0 <- min(which(regime_hardoff_2021[[6]] > prob_threshold))
cpt_h1 <- min(which(regime_hardoff_2122[[6]] > prob_threshold))

cpt_n12 <- min(which(regime_nitto_212q[[6]] > prob_threshold))
cpt_n14 <- min(which(regime_nitto_214q[[6]] > prob_threshold))

cpt_a12 <- min(which(regime_asahikasei_212q[[6]] > prob_threshold))
cpt_a14 <- min(which(regime_asahikasei_214q[[6]] > prob_threshold))




# generating sample path for testing CI
true_price <- function(S0, par, dt, duration, n_samples = 10^5) {
    r <- par[2] - par[1]^2 / 2
    diff_duration <- -diff(duration) * dt
    n_data <- length(duration) - 1
    log_sample <- matrix(rnorm(n_data * n_samples), ncol = n_samples, nrow = n_data)* par[1] * sqrt(diff_duration)
                            + r * diff_duration
    true_terminal_samples <- S0 * exp(colSums(log_sample))
    return(true_terminal_samples)
}



# 2nd step: Kalman filter
kalman_filter <- function(stock_data, dividend, dt, co_sigma, gamma, start_point, par){ # par = c(sigma_epsilon, delta, sigma_eta)
    stock_price <- stock_data[[3]][start_point:length(stock_data[[3]])]
    stock_duration <- stock_data[[2]][start_point:length(stock_data[[3]])]
    n <- length(stock_price) - 1
    pre_price <- stock_price[n + 1]
    post_price <- pre_price - dividend
    initial <- stock_price[1]
    pred_beta <- rep(0, n)
    filter_beta <- rep(0, n + 1)
    pred_sigma <- rep(0, n)
    filter_sigma <- rep(0, n + 1)
    pred_data <- rep(0, n)
    pred_error <- rep(0, n)
    pred_error_var <- rep(0, n)
    filter_beta[1] <- log(initial)
    filter_sigma[1] <- co_sigma
    normalized_path <- log(stock_price) - (stock_duration[1] - stock_duration) / stock_duration[1] * (log(pre_price) - log(post_price)) -
            par[2] * (stock_duration[1] - stock_duration) * dt
    for(i in 1:n){ # Kalman Filter
        pred_beta[i] <- gamma * filter_beta[i]
        pred_sigma[i] <- gamma^2 * filter_sigma[i] + par[3]^2
        pred_data[i] <- pred_beta[i]
        pred_error[i] <- normalized_path[i+1] - pred_data[i]
        pred_error_var[i] <- pred_sigma[i] + par[1]^2
        filter_beta[i+1] <- pred_beta[i] + pred_sigma[i] / pred_error_var[i] * pred_error[i]
        filter_sigma[i+1] <- pred_sigma[i] * (1 - pred_sigma[i] / pred_error_var[i]) + par[3]^2
    }
    pred_alpha <- pred_beta + par[2] * (stock_duration[1] - stock_duration[-1]) * dt
    pred_alpha <- c(log(stock_data[[3]][1:(start_point - 1)]), pred_alpha)
    filter_alpha <- filter_beta  + par[2] * (stock_duration[1] - stock_duration) * dt
    filter_alpha <- c(log(stock_data[[3]][1:(start_point - 1)]), filter_alpha)
    return(list(pred_alpha, pred_sigma, pred_data, pred_error, pred_error_var, filter_alpha, filter_sigma))
}

kalman_optimize <- function(stock_data, dividend, dt, co_sigma, gamma, start_point) {
```

```
    return(function(par){
        filtered <- kalman_filter(stock_data, dividend, dt, co_sigma, gamma, start_point, par)
        likelihood <- sum(log(dnorm(filtered[[4]], 0, sqrt(filtered[[5]]))))
        return(likelihood)
        }
    )
}

# setting for Kalman filter
gamma <- 1
co_sigma <- 1000
sigma_eta <- 0.3*sqrt(dt)
cond <- c(0, -1, 0)
mat <- diag(c(1, 0, 1))

init_y20 <- c(regime_yucho_2021[[1]][1] * sqrt(dt), regime_yucho_2021[[1]][2], sigma_eta)
init_y21 <- c(regime_yucho_2122[[1]][1] * sqrt(dt), regime_yucho_2122[[1]][2], sigma_eta)

init_h20 <- c(regime_hardoff_2021[[1]][1] * sqrt(dt), regime_hardoff_2021[[1]][2], sigma_eta)
init_h21 <- c(regime_hardoff_2122[[1]][1] * sqrt(dt), regime_hardoff_2122[[1]][2], sigma_eta)

init_n12 <- c(regime_nitto_212q[[1]][1] * sqrt(dt), regime_nitto_212q[[1]][2], sigma_eta)
init_n14 <- c(regime_nitto_214q[[1]][1] * sqrt(dt), regime_nitto_214q[[1]][2], sigma_eta)

init_a12 <- c(regime_asahikasei_212q[[1]][1] * sqrt(dt), regime_asahikasei_212q[[1]][2], sigma_eta)
init_a14 <- c(regime_asahikasei_214q[[1]][1] * sqrt(dt), regime_asahikasei_214q[[1]][2], sigma_eta)

est_path <- function(init, stock_data, dividend, dt, co_sigma, gamma, start_point){
    result <- constrOptim(init, kalman_optimize(stock_data, dividend, dt, co_sigma, gamma, start_point), grad = NULL,  ui=mat, ci = cond, control = list(fnscale = -1))
    opt_par <- result$par
    kalman_est <- kalman_filter(stock_data, dividend, dt, co_sigma, gamma, start_point, opt_par)
    est_path <- exp(kalman_est[[6]])
    est_par <- c(opt_par[1] / sqrt(dt), opt_par[3] / sqrt(dt),  opt_par[2])
    plot(stock_data[[1]], stock_data[[3]], type="l", lty=1, ylim=c(0.9*min(stock_data[[3]]), 1.1*max(stock_data[[3]])), ann=F)
    par(new=T)
    plot(stock_data[[1]], est_path, type="l", lty=2, ylim=c(0.9*min(stock_data[[3]]), 1.1*max(stock_data[[3]])), ann=F)
    legend("topleft", legend = c("data", "hidden"), lty = 1:2)
    return(list(est_par, est_path, kalman_est))
}

k_res_y20 <- est_path(init_y20, yucho_2021, div_yucho[1], dt, co_sigma, gamma, cpt_y0)
k_res_y21 <- est_path(init_y21, yucho_2122, div_yucho[2], dt, co_sigma, gamma, cpt_y1)

k_res_h20 <- est_path(init_h20, hardoff_2021, div_hardoff[1], dt, co_sigma, gamma, cpt_h0)
k_res_h21 <- est_path(init_h21, hardoff_2122, div_hardoff[2], dt, co_sigma, gamma, cpt_h1)


k_res_n12 <- est_path(init_n12, nitto_212q, div_nitto[1], dt, co_sigma, gamma, cpt_n12)
k_res_n14 <- est_path(init_n14, nitto_214q, div_nitto[2], dt, co_sigma, gamma, cpt_n14)

k_res_a12 <- est_path(init_a12, asahikasei_212q, div_asahikasei[1], dt, co_sigma, gamma, cpt_a12)
k_res_a14 <- est_path(init_a14, asahikasei_214q, div_asahikasei[2], dt, co_sigma, gamma, cpt_a14)


# final step: back tests
# estimate long-term volatility
archive_price <- yucho_1520[[3]]
archive_duration <- yucho_1520[[2]]
archive_sample <- diff(log(archive_price))
archive_delta <- -diff(archive_duration)*dt
archive_ll <- function(sample, delta){
    return(function(par){
        r <- par[1]-par[2]^2/2
        return(sum(log(dnorm(sample, r*delta, par[2]*sqrt(delta)))))
    }
    )
}

result <- optim(par=c(mean(archive_sample*archive_delta), sd(archive_sample/sqrt(archive_delta))), fn=archive_ll(archive_sample, archive_delta), control=list(fnscale=-1))
archive_sigma <- result$par[2]


# real-time valuing
time_point <- c(1229, 1240, 1251, 1262, 1273, 1284, 1295)
N_samples <- 10^4
```

```
N_div <- 500
significance <- 0.95
after_fall <- matrix(0, ncol=2, nrow=length(time_point))
terminal_date <- as.Date("2021-03-29")

# generating sample path for forecasting
sample_path <- function(S0, par, T, N_div=10^5, N_samples=10^5){ # T is yearly rate
    r <- par[2]-par[1]^2/2
    dt <- T/N_div
    log_sample <- matrix(rnorm(N_div*N_samples), ncol=N_samples, nrow=N_div)*par[1]*sqrt(dt)+r*dt
    true_terminal_samples <- S0*exp(t(apply(log_sample, 2, cumsum)))
    return(true_terminal_samples)
}


forecast_results <- list()

for(i in seq_along(time_point)){
    parameter_1 <- numeric(2)
    parameters_optim <- matrix(0, ncol=3, nrow=N_samples)
    stock_data <- extract_data(yucho, c(1135, time_point[i])) # list type
    N_days <- length(stock_data[[1]])
    # EM algorithm
    list_regime <- em_function(stock_data, dt, p_12, p_21, m, epsilon, init_par1, init_par2)
    parameter_1 <- list_regime[[1]]
    part_index <- min(which(list_regime[[6]] > prob_threshold))
    if (part_index == Inf){part_index=N_days}
    res_day <- as.numeric(terminal_date-stock_data[[1]][N_days])
    sample_dt <- res_day*dt/N_div
    samples <- cbind(t(matrix(stock_data[[3]][part_index:N_days], ncol=N_samples, nrow=length(stock_data
      [[3]][part_index:N_days]))), sample_path(stock_data[[3]][N_days], parameter_1, res_day*dt, N_div, N_
      samples))
    sample_duration <- c(as.numeric(terminal_date - stock_data[[1]][part_index:N_days]), rev(0:(N_div-1)*
      res_day/N_div))
    alpha_path <- matrix(0, ncol=length(samples[1,]), nrow=N_samples)

    # inits
    init_parameter <- c(list_regime[[1]][1] * sqrt(dt), list_regime[[1]][2], sigma_eta)

    for(j in 1:N_samples){
    k_res <- est_path(init_parameter, list(numeric(), sample_duration, samples[j, ]), div_yucho[1], dt, co_
      sigma, gamma, 1)
    parameters_optim[j, ] <- k_res[[1]]
    alpha_path[j, ] <- k_res[[2]][-1]
    }
    forecast_results[[i]] <- list(stock_data[[1]][N_days], alpha_path, parameter_1, parameters_optim,
      sample_duration)
    after_fall[i, ] <- c(mean(alpha_path[, N_div+1]), sd(alpha_path[, N_div+1])/sqrt(N_samples))
    print(i)
    print(Sys.time())
}
print(after_fall)
toc()
```

References

Cheang, Gerald HL and Carl Chiarella (2011). “Exchange options under jump-diffusion dynamics”. In: *Applied Mathematical Finance* 18.3, pp. 245–276.

Fortune, Peter et al. (1999). “Are stock returns different over weekends? A jump diffusion analysis of the weekend effect”. In: *New England Economic Review* 10, pp. 3–19.

Hainaut, Donatien and Franck Moraux (2019). “A switching self-exciting jump diffusion process for stock prices”. In: *Annals of Finance* 15.2, pp. 267–306.

Hamilton, James D. (1990). “Analysis of time series subject to changes in regime”. In: *Journal of Econometrics* 45.1, pp. 39–70. ISSN: 0304-4076. DOI: https://doi.org/10.1016/0304-4076(90)90093-9. URL: https://www.sciencedirect.com/science/article/pii/0304407690900939.

Hamilton, James Douglas (2020). *Time series analysis*. Princeton university press.

Hanson, Floyd B and John J Westman (2002). “Optimal consumption and portfolio control for jump-diffusion stock process with log-normal jumps”. In: *Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301)*. Vol. 5. IEEE, pp. 4256–4261.

Kabutan, 株探 (n.d.). **【特集】まだ間に合う**, *9* **月配当【高利回り】ベスト** *30* **「スタンダード他」編 割安株特集**. `https://kabutan.jp/news/marketnews/?b=n202209240146` (参照 2022-10-29).

Kim, Chang-Jin, Charles R Nelson, et al. (1999). “State-space models with regime switching: classical and Gibbs-sampling approaches with applications”. In: *MIT Press Books* 1.

Kou, Steven G (2002). “A jump-diffusion model for option pricing”. In: *Management science* 48.8, pp. 1086–1101.

Ma, Hui and Yang Li (2020). “Stock Price Jump-diffusion Process Model Based on Fractional Brownian Motion Theory”. In: *2019 3rd International Conference on Education, Economics and Management Research (ICEEMR 2019)*. Atlantis Press, pp. 374–378.

Maekawa, Koichi et al. (2008). “Jump diffusion model with application to the Japanese stock market”. In: *Mathematics and Computers in Simulation* 78.2-3, pp. 223–236.

Merton, Robert C (1976). “Option pricing when underlying stock returns are discontinuous”. In: *Journal of financial economics* 3.1-2, pp. 125–144.

Ramezani, Cyrus A and Yong Zeng (1998). “Maximum likelihood estimation of asymmetric jump-diffusion processes: Application to security prices”. In: *Available at SSRN 606361*.

Scott, Louis O (1997). “Pricing stock options in a jump-diffusion model with stochastic volatility and interest rates: Applications of Fourier inversion methods”. In: *Mathematical Finance* 7.4, pp. 413–426.

Simonato, Jean-Guy (2011). “Computing American option prices in the lognormal jump–diffusion framework with a Markov chain”. In: *Finance Research Letters* 8.4, pp. 220–226.

Tsay, Ruey S (2010). *Analysis of Financial Time Series*. John Wiley & Sons.

小西貞則, 越智義道, and 大森裕浩 (2008). **計算統計学の方法***:* **ブートストラップ・***EM* **アルゴリズム・***MCMC*. シリーズ予測と発見の科学 / 北川源四郎, 有川節夫, 小西貞則, 宮野悟編. 朝倉書店. ISBN: 9784254127850. URL: `https://books.google.co.jp/books?id=FIf1PAAACAAJ`.

沖本竜義 (2010). **経済・ファイナンスデータの計量時系列分析**. 朝倉書店.

第 3 章

災害債券市場における逆問題の数値的解法

三田 光星, 江上 雅彦*1

3.1 Introduction

災害債券 (CAT ボンドとも呼ばれる) は災害に関する指標に基づいて価値が決まる債券である. その指標の例として, 対象の災害にまつわる被害の総額などが挙げられる. 一般に根源的なリスクが災害であることにより, 災害債券はその他の資産と比較して市場リスクが小さいため, 市場リスクを軽減することを目的に投資家たちはしばしば災害債券をポートフォリオに加えている.

先行研究において Burnecki (2005) が複合ポアソン過程を用いて災害債券の評価のモデルを構築し, 以降さまざまな研究者が拡張や調整を行ってきた. 詳しくは Härdle and Cabrera (2010), Jarrow (2010), Z.-G. Ma and C.-Q. Ma (2013), Nowak and Romaniuk (2013) などを参照のこと. しかし, これらは損失の確率過程 $\{L_t\}$ をさまざまなデータによって推計し, $\{L_t\}$ によって災害債券の評価を行う. たとえば Burnecki (2005) ではデータ元として Property Claim Services の情報を用いて分布を特定し, その過程を用いて債券を評価するといった方法である. この場合, 公開されている被害の情報が全てではない場合があり, 保険会社のみが保持している情報もあると考えられるため, 投資家が正しくリスクを推計することは困難となることも多いと考えられる.

そのほかの方法として考えられるものはジャンプを伴う拡散過程を価格そのものに適用することである. ジャンプを伴う拡散過程は資産価格評価に頻繁に使われることが知られているが, 資産価格の動きを検証するためには同じ構造, あるいは似た構造のリスクを持つ過去のデータが必要である. 災害債券の数とそのリスクの性質からジャンプを伴う拡散過程を価格そのものに適用することは不適切であると考えられる.

以上の理由より, 価格のデータから災害債券の本来的なリスク (以下, $\{A_t\}$) を推計することは有用である. この論文では $\{A_t\}$ を価格のデータのみの状況において簡単な設定で推計し, その $\{A_t\}$ を用いて複雑な構造を持つ災害債券の評価が容易に行えることを検証する. 第 2 節では $\{A_t\}$ をジャンプのみからなる subordinator という一般的な設定の下で今回の推計方法について議論し, その中で単純な複合ポアソン過程について具体的に考える. 第 3 節では実際の災害債券の価格データを用いて, 複合ポアソン過程モデルにおける $\{A_t\}$ のパラメータと実現されたパスを推計する. 第 4 節では第 3 節で推計した $\{A_t\}$ を用いて, 同タイプのリスク構造を持つ複雑な災害債券の評価について確認を行う.

*1 (京都大学経済学研究科 教授 egami.masahiko.8x@kyoto-u.ac.jp)

3.2 一般的な議論

初めに Burnecki (2005) を元に, 満期が時刻 T のゼロクーポン災害債券の価格をモデル化する. $\tau = \inf\{s \in \mathbb{R}_+ \mid A_s > u\}$ は通過時刻, r はリスクフリーレート, $\{A_t\}$ は $A_0 = 0$, Lévy measure が μ の斉時的な subordinator でドリフトを持たない. また, $\mathcal{F}_t$ は $\{A_t\}$ から生成されるフィルトレーションとすれば, ゼロクーポン災害債券の価格は

$$\begin{aligned} V_t &= e^{-r(T-t)}\mathbb{E}[\mathbf{1}_{\{\tau>T\}} \mid \mathcal{F}_t] \\ &= e^{-r(T-t)}\mathbb{E}[\mathbf{1}_{\{A_T<u\}} \mid \mathcal{F}_t] \\ &= e^{-r(T-t)}\mathbb{P}(A_T < u \mid \mathcal{F}_t) \end{aligned} \tag{3.1}$$

となる. 2 つ目の等号は $\{A_t\}$ が subordinator であるため, t に関して増加であることから成立する. なおこのモデルは誘導型モデルと呼ばれ, デフォルトする可能性がある債券の評価に使われている. 誘導型モデルの詳細については Capiński and Zastawniak (2016) などを参照のこと.

(3.1) において, V_t の変動は連続的な要素と不連続な要素の 2 つからなる. 連続的な変動は満期が近づくことによる価値の上昇である一方で, 不連続な変動は実際の被害によって生じる価値の減少である. 災害債券における本質的な価値の変動は後者の不連続な変動によって起こると考えられるため, 以下では価格がジャンプしている点に主眼を置いて考えることとする.

A_t の推計方法を説明するために, 災害債券の価格の時系列データ $\{V_t\}$ とパラメータ $\boldsymbol{\theta}$ を所与とする. モデル (3.1) の性質上, $\{V_t\}$ が下方にジャンプしている点と $\{A_t\}$ が上方にジャンプしている点は一致することに注意して, 以下の手順で $\{V_t\}$ から $\{A_t\}$ を推計し, 理論価格 $\{\hat{V}_t\}$ を再現することを試みる.

1. V_t が下方にジャンプしている点を特定し, それらを $(t_1, t_2, \cdots)$ とする
2. $A_0 = 0$ を初期値として逐次的に A_{t_i} を計算し, 推計された $\{A_t\}$ を $\{\hat{A}_t\}$ とする
3. $\{\hat{A}_t\}$ と $\boldsymbol{\theta}$ を (3.1) に代入することで $\{\hat{V}_t\}$ を再現する

手順 1. については連続的な増加と不連続な減少の大きさを比較すると, 災害債券の性質により連続的な増加の大きさの方が小さいことが考えられる. したがって $dV_t < 0$ となっている点を抽出し $(t_1, t_2, \cdots)$ とする. また, 手順 3. は $\{\hat{A}_t\}$ と $\boldsymbol{\theta}$ を用いればモデルの式より計算されるため, 以下では手順 2. について詳述する.

$\{\hat{A}_t\}_{t\in[0,t_n]}$ が所与であるとする. (t_n, t_{n+1}) の間の $\{A_t\}$ についてはジャンプがないため $A_t = A_{t_n}, t \in (t_n, t_{n+1})$ である. また $t = t_{n+1}$ においてジャンプをしていることから

$$\begin{aligned} dV_{t_{n+1}} &= e^{-r(T-t_{n+1})}[\mathbb{P}(A_T < u \mid A_{t_{n+1}}) - \mathbb{P}(A_T < u \mid A_{t_n})] \\ &= e^{-r(T-t_{n+1})}[\mathbb{P}(A_T < u \mid A_{t_n} + z) - \mathbb{P}(A_T < u \mid A_{t_n})] \end{aligned}$$

である. ここで $\{A_t\}$ は斉時的な確率過程であることからモデルのパラメータは変わらず, 変わるのは閾値 u までの距離 $u - A_t$ の値のみであることに注意すれば,

$$\begin{aligned} dV_{t_{n+1}} &= e^{-r(T-t_{n+1})}\left[\int_0^{u-A_{t_{n+1}}} \mathbb{P}(A_T \in da) - \int_0^{u-A_{t_n}} \mathbb{P}(A_T \in da)\right] \\ &= -e^{-r(T-t_{n+1})}\int_{u-A_{t_{n+1}}}^{u-A_{t_n}} \mathbb{P}(A_T \in da) \end{aligned}$$

である. これより A_T の分布がわかればデータから計算される dV_t の値と照らし合わせることで $A_{t_{n+1}}$ の値, つまり t_{n+1} におけるジャンプの大きさを推計することが可能である. $\mathbb{P}(A_T \in da)$ が明示的に計算できない場合は subordinator の性質 (cf. Çınlar (2011) 7.7.2 式) より

$$\begin{aligned}\int_{\mathbb{R}_+} e^{-\beta a}\mathbb{P}(A_T \in da) &= \mathbb{E}[e^{-\beta A_T} \mid A_{t-}] \\ &= \mathbb{E}[e^{-\beta(A_{t-}+A_T-A_{t-})} \mid A_{t-}] \\ &= e^{-\beta A_{t-}}\mathbb{E}[e^{-\beta(A_T-A_{t-})} \mid A_{t-}] \\ &= \exp\left[-\beta A_{t-} - (T-t)\left(\delta\beta + \int_{\mathbb{R}_+}\mu(dx)(1-e^{-\beta x})\right)\right] \end{aligned} \tag{3.2}$$

となることを用いて, (3.2) を β の関数と見て逆ラプラス変換を行えば $\mathbb{P}(A_T \in da)$ は数値的に計算される. 数値的な逆ラプラス変換については例えば Zakian の方法 (cf. Halsted and Brown (1972)) によって簡単かつ十分高い精度で求められることがわかっている.

以上のように行えば, $\boldsymbol{\theta}$ の関数として $\{\hat{A}_t\}$ と $\{\hat{V}_t\}$ のパスを表現することが可能となる. この $\{\hat{V}_t\}$ と実際の $\{V_t\}$ の誤差が一番小さくなるような $\boldsymbol{\theta}$ をパラメータの推定値 $\hat{\boldsymbol{\theta}}$ とすることで, 災害債券が対象にしているリスク $\{A_t\}$ の市場における動きが捉えられる. たとえば二乗誤差をとって, $\hat{\boldsymbol{\theta}} = \arg\min_{\Theta} \sum_t |V_t - \hat{V}_t(\boldsymbol{\theta})|^2$ とするなどが考えられ, 今回の分析においてはこの二乗誤差を基準に推計した.

ここまでの議論はジャンプのみで構成される斉時的な subordinator の仮定の下で行われている. したがってジャンプの大きさやジャンプの頻度に関しての自由度が高く, 非常に広いクラスの確率過程を扱うことができ, ノンパラメトリックな設定を考えることも可能となる. 以降はその中でも簡単な設定として単純な複合ポアソン過程を仮定して話を進めていく.

3.2.1 今回の設定: 複合ポアソン過程

$\{A_t\}$ はパラメータ λ のポアソン過程とパラメータ η の指数分布による複合ポアソン過程, つまり独立な指数分布に従う確率変数列 $\{Y_n\}$ と, $\{Y_n\}$ と独立なポアソン過程 N_t を用いて $A_t = \sum_{n=1}^{N_t} Y_n$ と表現されるような過程であるとする. この場合, 指数分布に従う独立な確率変数の和がガンマ分布になることと, 複合ポアソン過程のマルコフ性を用いれば, 分布は逆ラプラス変換を用いることなく特定が可能で

$$\begin{aligned} V_t &= e^{-r(T-t)}\mathbb{P}(A_T < u \mid \mathcal{F}_t) \\ &= e^{-r(T-t)}\sum_{n=0}^{\infty}\mathbb{P}(N_{(t,T]} = n)\mathbb{P}(A_T < u \mid N_{(t,T]} = n, A_t) \\ &= e^{-r(T-t)}\left[e^{-\lambda(T-t)} + \sum_{n=1}^{\infty}\frac{\{\lambda(T-t)\}^n e^{-\lambda(T-t)}}{n!}\frac{\gamma(n,\eta(u-A_t))}{\Gamma(n)}\right] \end{aligned} \tag{3.3}$$

と明示的に表せる. ここで $N_{(t,T]} = N_T - N_t$ であり, $\gamma(n,\cdot) = \int_0^{\cdot} t^{n-1}e^{-t}dt$ の不完全ガンマ関数である. 分布が明示的に示されることより,

$$dV_t = e^{-r(T-t)}\sum_{n=1}^{\infty}\frac{\{\lambda(T-t)\}^n e^{-\lambda(T-t)}}{n!}\left[\frac{\gamma(n,\eta(u-\{A_{t-}+z\})) - \gamma(n,\eta(u-A_{t-}))}{\Gamma(n)}\right]$$

を z について解くことで各 t_n 時点での A_{t_n} のジャンプの大きさが推計できる. 分布関数の性質より右辺の [ ] で囲まれた部分は z に関して単調減少であることから, 一意かつ容易に解くことができ, 前述の逆ラプラス変換を数値的に行う方法よりも計算コストが小さい.

3.3 使用データとその特性

以下では地中海周辺の地震をリスクとする災害債券 *MedQuake.Ltd* の価格の日次データを使用した. 地震はハリケーンなどとは異なり季節性を考慮する必要がないと考えられるため, データの季節調整などが不要であり, 今回の分析に適しているデータといえる. Artemis によれば「2007 年 5 月から 2010 年 5 月までのトルコ・ギリシャ・イスラエル・ポルトガル・キプロスにおける地震」を対象に 100 万ドルの保証を行う債券であり, Thomson Reuters Datastream ® に記録されている全期間 (2008 年 11 月 3 日から 2010 年 4 月 23 日まで) のデータを取得し使用した. 便宜上満期は 2010 年 5 月 31 日としている. 該当地域では 2009 年 7 月 1 日にクレタ島で大地震が発生しており, 実際に図 3.1 を参照すれば 7 月 1 日付近で大きく価格が下がっていることが窺える. なお, 以下の分析では Datastream ® から取得した価格のデータを 100 で割ることで価格を 1 に正規化していることに注意する.

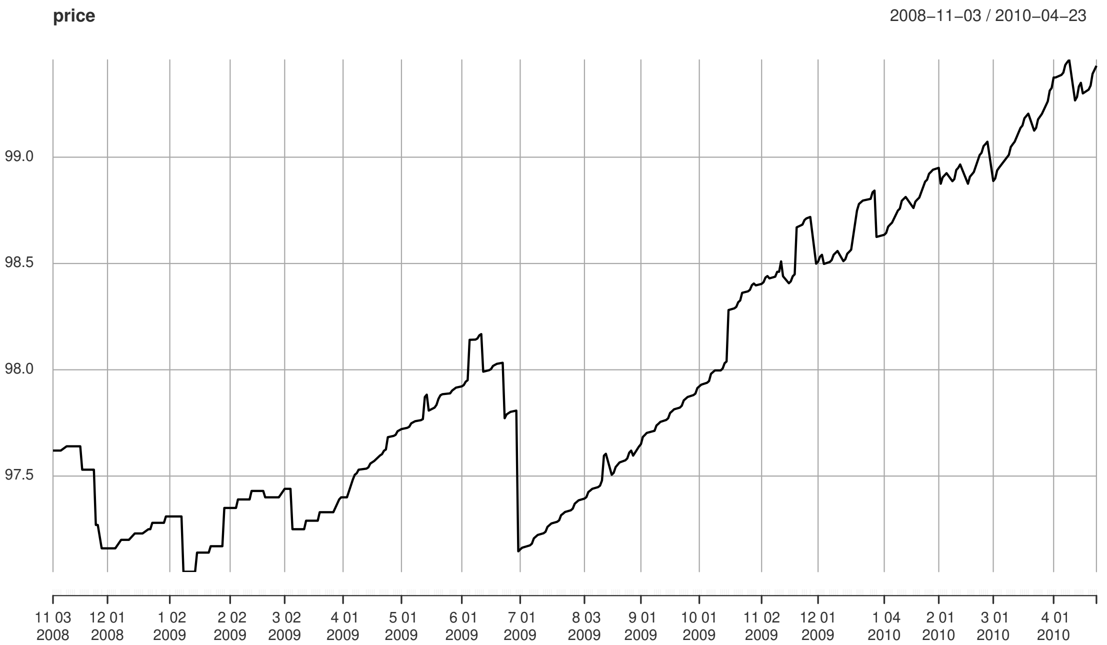

図 3.1　MidQuake.Ltd の価格推移

3.3.1 データの市場リスクに関する分析

一般に災害債券は市場リスクが小さいものとされているが, 実際に市場リスクが小さいかを確認しておく必要がある. この節では前節のモデルに立ち戻り, 今回使用する債券の価格データの市場リスクが小さいかを確認する. そのためにまず (3.3) の割引因子 $e^{-r(T-t)}$ を除いた $V^{\star}$ について離散化し, テイラー展開を用いて差分方程式を作る. ここでは $\{A_t\}$ による債券の価値の変化に注目するために割引因子を除いた.

使用データが日次データであり, 年率の時間幅 $\varDelta$ は十分小さいと考えられるため, 以下では期間 $[t, t+\varDelta]$ の

間にはジャンプが高々 1 度しか起こらないと仮定する. このとき $\mathbb{P}(N(t, t+\Delta] = 0) = e^{-\lambda\Delta}, \mathbb{P}(N(t, t+\Delta] = 1) = 1 - e^{-\lambda\Delta}$ であることに注意する.

$N(t, t+\Delta] = 0$ かつ $N(t, T] = k$, つまり $A_{t+\Delta} = A_t$ であるとき, t と $t+\Delta$ における生存確率は $F_\Gamma(u - A_t \mid k)$ である. したがって $N(t, t+\Delta] = 0$ である場合,

$$\begin{aligned}\Delta V_t^\star &= V_{t+\Delta}^\star - V_t^\star \\ &= \sum_{k=0}^{\infty} (\mathbb{P}(N(t+\Delta, T] = k) - \mathbb{P}(N(t, T] = k)) F_\Gamma(u - A_t \mid k) \\ &=: \mu_t(\Delta)\end{aligned}$$

が成立する. $dV_t^\star$ は満期までに起こるジャンプの頻度 $\lambda(T-t)$ が期間が短くなって下がることで, 結果的に生存確率が上がるために生じる連続的な増加のみで構成されている. 一方で期間中にジャンプが 1 度発生する場合, つまり $N(t, t+\Delta] = 1$, $N(t, T] = k+1$ $(N(t+\Delta, T] = k)$ である場合, t における生存確率は $F_\Gamma(u - A_t \mid k+1)$ であり, $t+\Delta$ における生存確率は $F_\Gamma(u - A_{t+\Delta} \mid k)$ である. したがって $N(t, t+\Delta] = 1$ であるときの価格の変化は

$$\begin{aligned}\Delta V_t^\star &= \sum_{k=0}^{\infty} [\mathbb{P}(N(t+\Delta, T] = k) F_\Gamma(u - A_{t+\Delta} \mid k) - \mathbb{P}(N(t, T] = k+1) F_\Gamma(u - A_t \mid k+1)] \\ &=: \tilde{J}_t(\Delta)\end{aligned}$$

$\tilde{J}_t$ は連続的な増加と不連続な減少の両方の要素を持っているため, 複雑な構造となっている. したがって一般には価格の増減について情報はないが, 災害債券の性質よりこの場合の価格は大きく減少すると考えられる.

線形の表現を行うために $\Delta = 0$ の周辺で $e^{-\lambda\Delta}\mu_t(\Delta)$ と $(1 - e^{-\lambda\Delta})\tilde{J}_t(\Delta)$ をテイラー展開する. $\mu_t(\Delta)$ と $\tilde{J}_t(\Delta)$ は C^∞ に属していることからテイラー展開可能であることに注意する. 表記を簡単にするために $p_k(t, A_t)$ と $a_m(t)$, $b_m(t, Z_t)$ を以下のように定義する.

$$\begin{aligned}p_k(t, A_t) &:= \frac{e^{-\lambda(T-t)}[\lambda(T-t)]^k}{k!} F_\Gamma(u - A_t \mid k) \\ a_m(t) &:= \sum_{k=m}^{\infty} \frac{(-1)^m p_k(t, A_t)_k C_m}{(T-t)^m} - \left[\sum_{k=0}^{\infty} p_k(t, A_t)\right] \frac{(-\lambda)^m}{m!} \\ b_m(t, Z_t) &:= \sum_{k=0}^{\infty} \sum_{n=0}^{k\wedge(m-1)} \frac{(-1)^n p_k(t, A_t + Z_t) \lambda^{m-n}{}_k C_n}{(T-t)^n (m-n)!} + \left[\sum_{k=0}^{\infty} p_{k+1}(t, A_t)\right] \frac{(-\lambda)^m}{m!} \\ &= \sum_{k=0}^{\infty} \sum_{n=0}^{k\wedge(m-1)} \frac{(-1)^n p_k(t, A_t + Z_t) \lambda^{m-n}{}_k C_n}{(T-t)^n (m-n)!} \\ &\quad + \left[\sum_{k=0}^{\infty} p_k(t, A_t)\right] \frac{(-\lambda)^m}{m!} - e^{-\lambda(T-t)} \frac{(-\lambda)^m}{m!}\end{aligned}$$

ここで Z_t は $[t, t+\Delta)$ におけるジャンプであり, モデルの設定からパラメータ η の指数分布に従うジャンプである. このとき

$$\begin{aligned}e^{-\lambda\Delta}\mu_t(\Delta) &= \sum_{k=0}^{\infty} \frac{e^{-\lambda(T-t)}[\lambda(T-t)]^k}{k!} F_\Gamma(u - A_t \mid k) \left[\sum_{l=0}^{k} \frac{(-1)^l {}_k C_l}{(T-t)^l} \Delta^l - \sum_{m=0}^{\infty} \frac{(-\lambda)^m}{m!} \Delta^m\right] \\ &= \sum_{m=1}^{\infty} a_m(t) \Delta^m = a_1(t)\Delta + o(\Delta)\end{aligned}$$

$$
\begin{aligned}
(1-e^{-\lambda\Delta})\tilde{J}_t(\Delta) = \sum_{k=0}^{\infty}\Bigg[& \frac{e^{-\lambda(T-t)}[\lambda(T-t)]^k}{k!}F_\Gamma(u-A_t-Z_t \mid k) \\
& \times \sum_{l=0}^{\infty}\sum_{n=0}^{k}\frac{\lambda^{l+1}}{(l+1)!}\left(\frac{-1}{T-t}\right)^n {}_kC_n\Delta^{l+n+1}\Bigg] \\
& - \sum_{k=0}^{\infty}\frac{e^{-\lambda(T-t)}[\lambda(T-t)]^{k+1}}{(k+1)!}F_\Gamma(u-A_t \mid k+1)\left[-\sum_{l=0}^{\infty}\frac{(-\lambda)^{l+1}}{(l+1)!}\Delta^{l+1}\right] \\
= \sum_{m=1}^{\infty} & b_m(t,Z_t)\Delta^m = b_1(t,Z_t)\Delta + o(\Delta)
\end{aligned}
$$

これより Δ について整理を行えば

$$
\begin{aligned}
\Delta V_t^{\star} &= e^{-\lambda\Delta}\mu_t(\Delta) + (1-e^{-\lambda\Delta})\tilde{J}_t(\Delta) \\
&= (a_1(t) + b_1(t,Z_t))\Delta + o(\Delta) \\
&= \left(\lambda e^{-\lambda(T-t)} - \frac{1}{T-t}\sum_{k=1}^{\infty}kp_k(t,A_t) + \sum_{k=0}^{\infty}\lambda p_k(t,A_t+Z_t)\right)\Delta + o(\Delta)
\end{aligned} \tag{3.4}
$$

と表現できる.

以上のように微小区間におけるジャンプの有無によって価格の変動を差分方程式として表すことで, 確率微分方程式の導出が可能である. 差分方程式の両辺を $V^{\star}$ で割り, 市場リスクとして独立なブラウン運動 σW_t を加えると

$$
\frac{\Delta V_t^{\star}}{V_t^{\star}} = \frac{(a_1(t) + b_1(t,Z_t))}{V_t}\Delta + \sigma W_\Delta + o(\Delta)
$$

という形で変動を方程式を表すことができる. この σ を推定し, 十分小さいかを検証するために, Aıt-Sahalia (2004) の Proposition 1 の $r < 2$ の結果を用いる. 詳細については付録を確認のこと.

$$
\begin{aligned}
M_a(\Delta,\theta,r) :=& \mathbb{E}[\,|\Delta V_t/V_t - \mathbb{E}[\Delta V_t/V_t]\,|^r] \\
=& \pi^{-1/2}2^{r/2}\Gamma\left(\frac{1+r}{2}\right)\sigma^r\Delta^{r/2} + o(\Delta^{r/2}).
\end{aligned}
$$

このモーメント条件に $r = \frac{1}{2}$ から推定される年率の市場リスクは $\hat{\sigma} = 0.003633533$ であるので, 十分に市場リスクが小さいことが確認できる.

3.3.2 モデルによる $\{\hat{A}_t\}$ の推計

市場リスクが十分に小さいことが確認されたため, 第 2 節の (3.3) で議論した内容をもとにデータ $\{V_t\}$ から $(\hat{\lambda},\hat{\eta})$ を推定, $\{\hat{A}_t\}$ と $\{\hat{V}_t\}$ を推計した. 債券の閾値は $u = 100$, データポイント間の期間の幅は $dt = 1/365$ に日数をかけたもの, リスクフリーレートには *MedQuake.Ltd* のデータの期間におけるポンド翌日物平均金利 (SONIA) の平均の値 $r = 0.00699612626262626$ を使用した. また, 本来であれば $dV_t = V_t - V_{t-}$ によってパラメータの推定を行うべきであるが, V_{t-} のデータは存在していない. 代替案として $V_{t-} = e^{r\Delta}V_{t-1}$ を使う, あるいは $V_{t-} = V_{t-1}$ を使うことが考えられる. 今回はリスクフリーレートが外部から決定する時間に依存しない定数であるが, 一般的なリスクフリーレートを設定する場合に, 前者を計算することで発生する誤差について考慮すれば, 後者を用いた場合に発生する誤差の方が構造を把握しやすい. また微小区間 $(t-\Delta, t]$ における連続的な増加については十分小さいと考えられるため, 今回は後者を採用し, $dV_t = V_t - V_{t-1}$ として推定を行った.

パラメータの推定結果は $(\hat{\lambda}, \hat{\eta}) = (0.74531781, 0.05928405)$ であり, 推計された $\{\hat{A}_t\}$ と $\{\hat{V}_t\}$ は図 3.2, 3.3 の通りである． 図 3.3 を見ると, 単純な設定と単調増加な複合ポアソン過程という簡単なモデルであるにも関わらず十分な精度で推計できていることがわかる. よって以下では乖離部分について検討する. まず最初にジャンプするまでについてはモデルと実際に市場で取引を開始した際の価格の間に乖離があるが, これは当初発行者が投資家たちよりもリスクを過小評価していたと考えられ, 市場で取引するうちに正しい水準に落ち着いたと考えることが自然である. その他の実データと乖離している部分について, 特に後半は過小評価になっていることが確認できるが, これはモデルの設定上 $\{V_t\}$ が大きく上にジャンプしている点を捉えられていないことから発生している. このようなモデルの外にあるジャンプについては次小節で考慮する.

3.3.3　実データとより適合する $\{\tilde{A}_t\}$ の推計

災害債券の価格はそのリスクの性質上, 災害が起こった直後に大きく下落をするが, 時間が経過した後に一定の水準まで価格が戻るという事象がしばしば観測される. その大きな要因として被害に対する過大な評価が時間経過によって修正されることが考えられる. これは 1 回の災害の被害の大きさは災害発生時点 t_n において可測ではなく, 市場が正しく被害の規模を把握するには時間がかかるためである. 今回使用した *MedQuake.Ltd* についても 10 月 16 日に価格が上方にジャンプするという形でこのような調整は現れている. しかし, このような調整プロセスは (3.3) のような簡単なモデルだけではなく, (3.1) のような一般的な設定においても, subordinator が単調増加であることから捉えることができない.

ただし, 調整は災害債券の本質的な価格変動ではない. 調整は被害の全貌が見えない状況で過大に被害を評価している結果に過ぎないため, 本来は調整後の値が発生した災害の被害にあたる. したがって災害債券の価格モデルについては変わらず $\{A_t\}$ が斉時的な subordinator であり, (3.1) で評価されると投資家たちは考えているという前提で, 以下のような手順によって $\{V_t\}$ の上方ジャンプを含めた $\{A_t\}$ の動きを捉える.

1. $\{V_t\}$ が上方にジャンプしている点と下方にジャンプしている点を抽出する
2. 抽出した点 $\{s_n\}$ において, $\hat{V}_t - V_t$ と推定したパラメータ $\hat{\boldsymbol{\theta}}$ から $\tilde{A}_{s_n}$ を求める
3. $t \in [s_n, s_{n-})$ において $\tilde{A}_t = \tilde{A}_{s_n}$ とする

下方ジャンプについても調整を行うのは, 3.2 節においてデータから計算される dV_t の値を $V_t - V_{t-1}$ としたことによる誤差が小さいながらも発生しているからである. この方法はジャンプしている点 $\{t_n\}$ に関してのみ調整を行うことから, $t \in [0, T] \backslash \{t_n\}$ において, 必ずしも $\{\tilde{V}_t\}$ が $\{V_t\}$ を正しく説明できる保証がないため, 以下ではこのような調整によって $\{\tilde{V}_t\}$ が $\{\hat{V}_t\}$ より正確に $\{V_t\}$ を再現できているかを検証した.

$\{A_t\}$ の下方ジャンプの点は上方ジャンプのときと異なり, 連続的に増加している部分のことを考えると単純に $dV_t > 0$ とすることはできない. また, (3.4) より明らかに連続的な増加は A_t の値に依存するが, 下方ジャンプのことを考慮していない $\hat{A}_t$ をそのまま使用することは不適切である. したがって今回は $dV_t \geq 0$ となっているポイントについて前後 15 ポイントの移動平均 μ_t をとり, $\{t \mid |dV_t - \mu_t| > 10^{-3}\}$ を $\{A_t\}$ が下方にジャンプしている点として採用した. なお, 前後 15 ポイントを取れない点については直近の μ_t の値を代入している.

推計された $\{\tilde{A}_t\}$ と $\{\tilde{V}_t\}$ は図 3.4, 3.5 の通りである． 3.2 節で確認した図 3.3 と図 3.5 を比較すると, 特に 2009 年 3 月 6 日から 7 月 1 日までと 2009 年 10 月 16 日以降の精度が上がっている. なお最初の価格データが下方ジャンプしている点において, 推計した $\{\tilde{V}_t\}$ は上にジャンプしているが, これは 3.2 節と同様の議論で, 市場と発行者の間のリスクに対する意識の乖離を調整しているために発生していると考えられる.

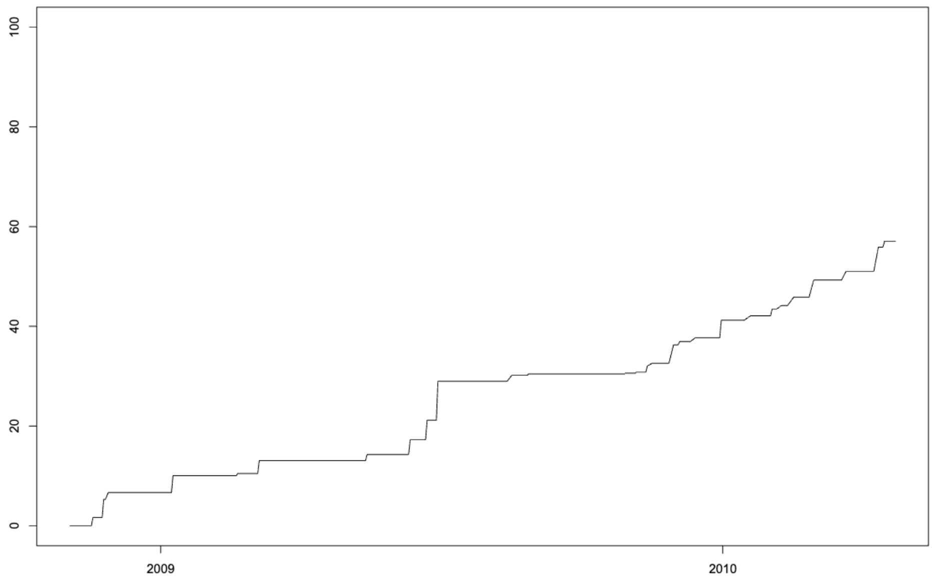

図 3.2　推計された $\{\hat{A}_t\}$ のパス

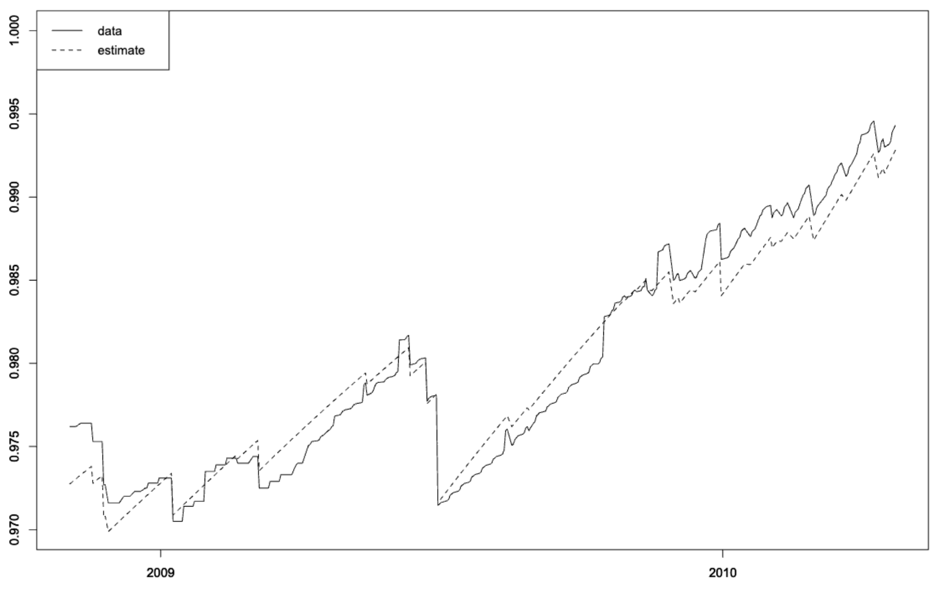

図 3.3　データ $\{V_t\}$ のパス (実線) と推計された $\{\hat{V}_t\}$ のパス (点線)

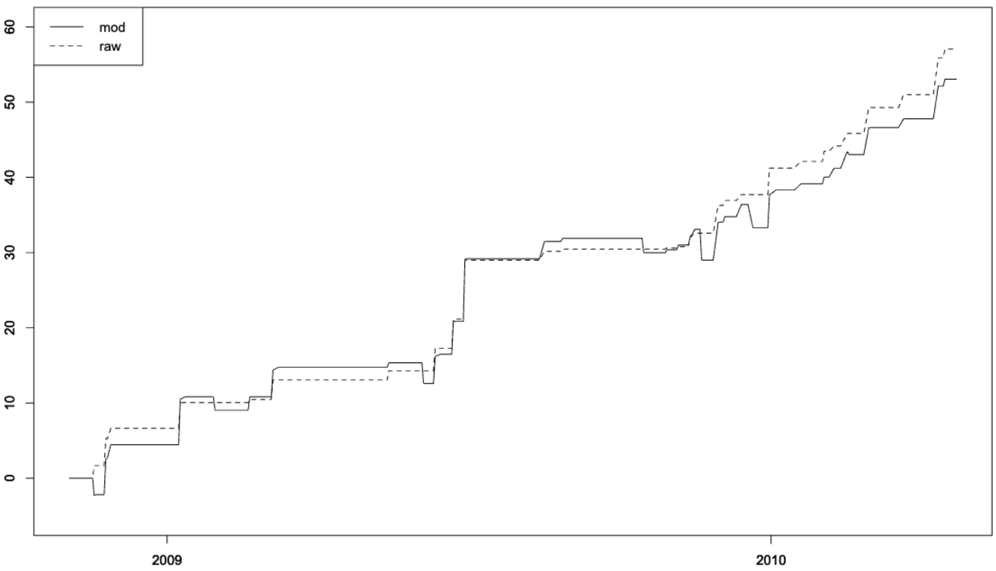

図 3.4　推計された $\{\tilde{A}_t\}$(実線) と $\{\hat{A}_t\}$(点線) のパス

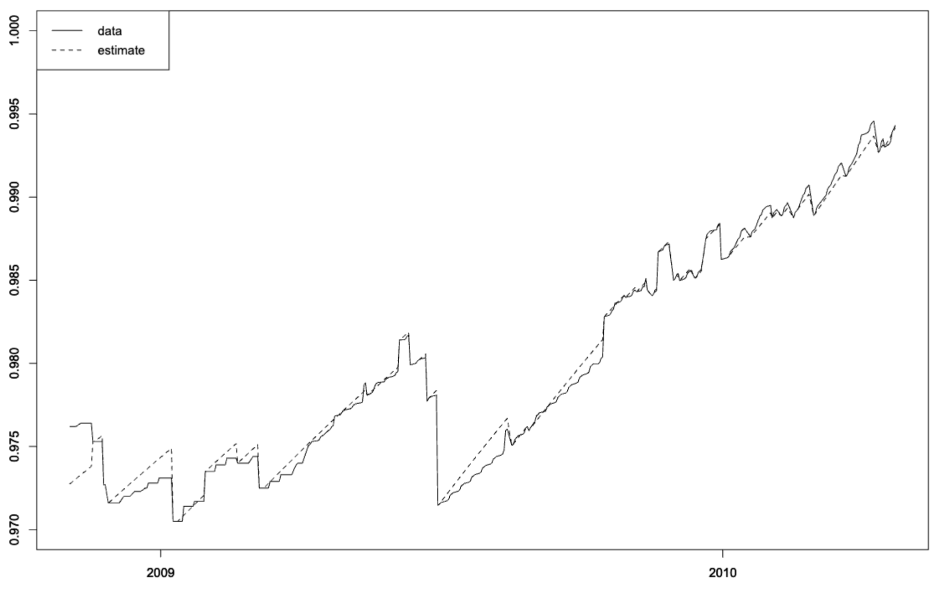

図 3.5　データ $\{V_t\}$ のパス (実線) と推計された $\{\tilde{V}_t\}$ のパス (点線)

3.4 Application

最後に複雑な構造を持つ災害債券の評価が簡単になることを確認する. 災害債券によくある構造の例として閾値が複数あり, 閾値を超えるごとに債券のペイオフが下がるラチェット型が知られている. 例としてWei, Liu, and Hou (2022) のようにトリガーに被害額と地震のマグニチュードの大きさの二種類を設定し, 満期までに片方のみが閾値を超えた場合は一部償還されるといった構造が挙げられる. これを参考にリスクが *MedQuake.Ltd* と同じ地中海の地震を対象とした以下のような構造の債券を考える.

1. 被害総額が u 万ドルを超えた場合, 額面の全額が償還されない
2. 1 回の被害額の大きさ $\{Z_n\}_{n\leq N_T}$ が 1 回でも $u/4$ を超えた場合は額面の半額が償還されない

ここで満期は時刻 T, 償還額は F ドル, リスクフリーレートは r とする. $\tau_z = \inf\{n \mid Z_n > u/4\}, \tau_a = \inf\{t \mid A_t > u\}$ とすればペイオフは以下のように定式化される.

$$\begin{aligned} V_T &= F * \mathbf{1}_{\{\tau_a > T\}}(\mathbf{1}_{\{\tau_z > T\}} + 0.5 * \mathbf{1}_{\{\tau_z \leq T\}}) \\ V_t &= e^{-r(T-t)} F * \mathbb{E}[\mathbf{1}_{\{\tau_a > T\}}(\mathbf{1}_{\{\tau_z > T\}} + 0.5\mathbf{1}_{\{\tau_z \leq T\}}) \mid \mathcal{F}_t] \\ &= e^{-r(T-t)}[F * \mathbb{P}(\tau_a > T, \tau_z > T \mid \mathcal{F}_t) + 0.5F * \mathbb{P}(\tau_a > T, \tau_z \leq T \mid \mathcal{F}_t)] \end{aligned}$$

今リスクの対象はリスクが *MedQuake.Ltd* と同じ地中海の地震であるので, 3 節で推計したパラメータ $(\hat{\lambda}, \hat{\eta}) = (0.74531781, 0.05928405)$ を用いた評価がモンテカルロ・シミュレーションによって可能である. 複合ポアソン過程のモンテカルロ・シミュレーションについては付録.B を参照のこと.

以下の表 3.1 は償還額 $F = 100$, リスクフリーレート $r = 0.001$, サンプルの数 $N = 10^5$, 分割の幅 $dt = 1/365$ として債券の現在価値を期間と閾値をごとにモンテカルロ法で求めた結果は以下である. 閾値 u に関して増加

閾値 u/期間 T	3 年	4 年	5 年
100	0.658352 (0.001182095)	0.6414242 (0.001160678)	0.5912812 (0.0012021)
200	0.8361029 (0.001038578)	0.8715767 (0.0008630181)	0.8776756 (0.0007878012)
300	0.8766361 (0.0009959085)	0.9277715 (0.0007404161)	0.9489782 (0.0005765699)
400	0.8866311 (0.0009820007)	0.9417405 (0.0007017396	0.966182 (0.0005042584)
500	0.8892831 (0.0009770631)	0.9452564 (0.0006894186)	0.9706297 (0.0004804709)

表 3.1　ラチェット型災害債券の価格 (標準誤差)

は設定より明らかであるが, 実際にモンテカルロ・シミュレーションによって計算された価格も u に関して増加であることは確認される. 一方で直観的には T が大きい, つまり期間が長くなればデフォルトする確率は上がると考えられるが, 閾値 u が 200 以上の場合は期間が長くなるにつれて価値が上昇する結果が出ている. これはパラメータ λT のポアソン分布の確率関数が $\mathbb{P}(N = n) = (\lambda T)^n \exp(-\lambda T)/n!$ であることが主に関係していると推測されるが, この傾向については「複雑な構造を持った災害債券の評価の検証」の範囲を逸脱するため今後の課題とする.

このように同種のリスクのゼロクーポン災害債券の価格のデータからパラメータを推計することで, 複雑な構造を持つ債券であってもシミュレーションによって債券価格を評価することができる.

3.5 結論と今後の展望

災害債券の価格モデルとして一般的な設定から出発し, 価格のデータから根源的なリスクの過程を推定する逆問題を数値的に解く方法を確立した. その上で最も単純な複合ポアソン過程の誘導型モデルを仮定し, 実際のデータを用いて逆問題を解いた. モデルと逆問題を解くことで推定したパスから計算される推計価格は元のデータとの整合性もあり, 十分な精度で逆問題の解を導くことが可能であることを確認した. この逆問題を解く方法を確立することで, 投資家と災害債券発行機関の間にある情報の非対称性は小さくなると考えられる. また市場で取り引きされた災害債券の価格データが存在すれば, 複雑な構造を持つ災害債券についてもシミュレーションによって評価が可能となり, さまざまな形のリスクをヘッジすることが可能となる.

一方で 3.3 節で確認した通り, 実際の価格のデータには本来災害債券のモデルでは考えられない上方のジャンプが存在しており, 逆問題を解くことで推計したパスよりもデータと整合するパスが存在している. 今回は増加の確率過程である subordinator を用いたため, 逆問題を解く前提に置いたモデルで捉えることができていない. このような調整の過程は subordinator と別のフレームワークで考える必要があり, たとえば価格が下方にジャンプした後に強度が上がるショットノイズのような点過程を用いたモデルが候補として上がるが, このような調整部分を含めたモデルの設定については今後の研究で取り組む予定である.

また, 今回の分析では複合ポアソン過程を仮定したが, より一般的にノンパラメトリックな設定を含むさまざまなモデル設定をすることでどのようなモデルが最も災害債券に適しているかを検証することも今後の課題である. 特に災害債券の性質として災害発生直後に激しい価格変動があるという点で, 時間に一様でない subordinator の設定での逆問題については重要な研究課題であると考えられる.

付録 3.A　Aıt-Sahalia (2004) 2.3 節について

Lepingle (1976) の議論により, ジャンプを伴う拡散過程 $\{X_t\}$ の r 次変分

$$ {}_r[X,X]_t = \operatorname*{plim}_{n\to\infty} \sum_{i=1}^{n} |X_{t_i} - X_{t_{i-1}}|^r $$

のジャンプが寄与する部分は, $r \in (0,2)$ のときに正規化を行うことで 0 となることがわかっている. この結果を用いて Barndorff-Nielsen and Shephard (2003) は $r \in (0,2)$ の場合, ${}_r[X,X]_t$ が拡散過程の部分にのみ依存することを示している. この結果を用いて Aıt-Sahalia (2004) は Proposition 1 として以下のようなモーメント条件を導出している.

$$ \begin{aligned} M_a(\varDelta, \boldsymbol{\theta}, r) :=& \mathbb{E}|Y_\varDelta - \mathbb{E}[Y_\varDelta]|^r \\ =& \frac{2^{\frac{r}{2}}}{\sqrt{\pi}} \Gamma\left(\frac{1+r}{2}\right) \sigma^r \varDelta^{\frac{r}{2}} + o(\varDelta^{\frac{r}{2}}) \end{aligned} $$

今回 Aıt-Sahalia (2004) と異なるのは差分が A_t に依存することとジャンプの部分であるが, このモーメントの式を A_t で条件付けた条件付きのモーメントとして考え, Barndorff-Nielsen and Shephard (2003) の結果を用いればこの Proposition 1 の式を適用することが可能である. なお (3.4) の議論には $V^\star$ という割引を行なっていないデータを用いたが, モーメント条件において期待値を引く操作を行うことから割引因子も同時に消えるため, $\varDelta V_t / V_t$ としてデータの対数差分を用いた.

付録 3.B　複合ポアソン過程のモンテカルロ・シミュレーション

複合ポアソン過程は独立な指数分布に従う確率変数列 $\{Y_n\}$ と $\{Y_n\}$ と独立なポアソン過程 N_t を用いて $A_t = \sum_{n=1}^{N_t} Y_n$ と表現されるような過程である. したがってシミュレーションを行う際にポアソン過程と指数分布の発生は別々に行うことが可能である. ここでシミュレーションを行う前にポアソン過程の性質について確認する.

Proposition 1. 強度 λ のポアソン過程 N_t は以下を満たす

1. 区間 $[0,T]$ におけるイベント数は強度 λT のポアソン分布に従う
2. 区間 $[0,T]$ におけるイベント数 n が所与の場合, イベントの発生時刻は区間区間 $[0,T]$ 内で独立な一様分布となる

証明の詳細は近江, 野村 (2019) などを参照のこと. この命題に基づいて $[0,T]$ における複合ポアソン過程を以下のような手順でシミュレーションを行う.

1. 強度 λT のポアソン分布に従う乱数を発生させる ($X^{(i)}$ とする)
2. 区間 $[0,T]$ の一様分布に従う乱数を $X^{(i)}$ 個発生させる (増加列になるよう並べ替えたものを $\tau_n^{(i)}$ とする)
3. 母数 η の指数分布に従う乱数を $X^{(i)}$ 個発生させる ($Y_n^{(i)}$ とする)
4. $N_t = \max\{n \mid \tau_n^{(i)} \leq t\}$ として, $A_t^{(i)} = \sum_{n=1}^{N_t} Y_n^{(i)}$ を計算する

付録 3.C　R 言語のスクリプト

Listing 3.1　Forecasting by R 4.2.1

```r

library(xts)

# settings
bond_data <- read.csv("[input␣path]")
date <- as.Date(bond_data[, 1])
bond_price <- bond_data[, 2] / 100 # normalize
av_sonia <- 0.00699612626262626 # SONIA risk free rate
dt <- 1/365
terminal_date <- as.Date("2010-05-31") # terminal date
duration <- as.numeric(terminal_date- date) * dt # annualized
jump_index <- which(diff(bond_price) < 0) + 1
jump_size <- diff(bond_price)[which(diff(bond_price) < 0)]
threshold <- 100
n_max <- 10^5 # for maximum # of jumps in [0, T]

# check market risk
logvalue=log(bond_price)
logdiff=diff(bond_price)/bond_price[1:(length(bond_price)-1)]
expect=mean(logdiff)
target=logdiff-expect
dt=1/365

# set moment function and estimate sigma
M_1=function(y, r, dt, sigma){ # only r<2
    moment=2^(r/2)*gamma((1+r)/2)*abs(sigma)^r*dt^(r/2)/sqrt(pi)
    if (sigma<0){return(mean(abs(y)^r-moment)^2-10^3)}
    else{return(mean(abs(y)^r-moment)^2)}
}
result=optimize(M_1, interval=c(0,1), y=target, r=1/2, dt=dt)
sigma=result$minimum



# 1st step
solving_jump_size <- function(dif_price, r, duration, threshold, n_max, lambda, eta) {
    return(function(raw_jump) {
        poisson <- dpois(0:n_max, lambda * duration)
        return(dif_price - exp(-r * duration) * sum(poisson * (pgamma(threshold - raw_jump, shape = 0:n_max
            , rate = eta) - pgamma(threshold, shape = 0:n_max, rate = eta))))
    })
}

parameter_optimize <- function(timepoint, price, duration, r, threshold, n_max) {
    dif_price <- diff(price)
    return(function(par) {
        raw_jump <- numeric(0)
        for (i in seq_along(timepoint)) {
            result <- try(uniroot(solving_jump_size(dif_price[timepoint[i]-1], r, duration[timepoint[i]],
                threshold - sum(raw_jump), n_max, par[1], par[2]), c(0, threshold - sum(raw_jump))))
            if (class(result) == "try-error") {
                root <- 0
            }else {
                root <- result$root
            }
            raw_jump <- c(raw_jump, root)
        }
        path_a <- numeric(length(price))
        path_a[timepoint] <- raw_jump
        path_a <- cumsum(path_a)
        theoretical <- exp(-r * duration) * colSums(t(dpois(t(matrix(0:n_max, nrow = n_max + 1, ncol =
            length(duration))), par[1] * duration)) * pgamma(t(matrix(threshold - path_a, ncol = n_max + 1,
            nrow = length(path_a))), shape = 0:n_max, rate = par[2]))
        return(sum((theoretical - price)^2))
    })
}

init_par <- c(length(jump_index) / duration[1], 0.46373) # lambda is estimated via 、
  dataeta is solution of model and initial data
ui <- matrix(c(1, 0, 0, 1), ncol = 2, nrow = 2)
```

```r
ci <- c(0, 0)
optim_result <- constrOptim(init_par, parameter_optimize(jump_index, bond_price, duration, av_sonia, threshold, n_max), grad = NULL,  ui = ui, ci = ci)

# regenerate path
path_generator <- function(timepoint, price, duration, r, threshold, n_max) {
    dif_price <- diff(price)
    return(function(par) {
        raw_jump <- numeric(0)
        for (i in seq_along(timepoint)) {
            result <- try(uniroot(solving_jump_size(dif_price[timepoint[i]-1], r, duration[timepoint[i]], threshold - sum(raw_jump), n_max, par[1], par[2]), c(0, threshold - sum(raw_jump))))
            if (class(result) == "try-error") {
                root <- 0
            }else { #exception handling
                root <- result$root
            }
            raw_jump <- c(raw_jump, root)
        }
        path_a <- numeric(length(price))
        path_a[timepoint] <- raw_jump
        path_a <- cumsum(path_a)
        theoretical <- exp(-r * duration) * colSums(t(dpois(t(matrix(0:n_max, nrow = n_max + 1, ncol = length(duration))), par[1] * duration)) * pgamma(t(matrix(threshold - path_a, ncol = n_max + 1, nrow = length(path_a))), shape = 0:n_max, rate = par[2]))
        return(list(theoretical, path_a))
    })
}
theoretical_value_func <- path_generator(jump_index, bond_price, duration, av_sonia, threshold, n_max)
estimate <- theoretical_value_func(optim_result$par)
lambda <- optim_result$par[1]
eta <- optim_result$par[2]
a_path_raw <- estimate[[2]]

# plot graphic
plot(date, bond_price, type="l", lty=1, ylim=c(0.97, 1), ann=F)
par(new=T)
plot(date, estimate[[1]], type="l", lty=2, ylim=c(0.97, 1), ann=F)
legend("topleft", legend = c("data", "estimate"), lty = 1:2)

# check points when A_t jumps downwards
error_threshold <- 10^-3
not_downward <- (1:length(bond_price))[-jump_index]
ma <- filter(diff(bond_price)[-(jump_index - 1)], rep(1, 30)/30)
ma[1:14] <- ma[15]
ma[328:342] <- ma[327]
upward_index <- not_downward[which(abs(ma - diff(bond_price)[-(jump_index + 1)]) > error_threshold)+1]

all_jumps_index <- sort(union(upward_index, jump_index))
dif <- bond_price[all_jumps_index] - estimate[[1]][all_jumps_index]
a_path <- estimate[[2]]
a_path_raw <- estimate[[2]]
all_jumps <- numeric()

for(i in seq_along(all_jumps_index)){
    res <- try(uniroot(solving_jump_size(dif[i], av_sonia, duration[all_jumps_index[i]], threshold - a_path[all_jumps_index[i]], n_max, lambda, eta), c(-1000, 1000)))
        if (class(res) == "try-error") {
            root <- 0
        }else { # exception handling
            root <- res$root
        }
    all_jumps <- c(all_jumps, root)
    affect_point <- all_jumps_index[i+1]
    if (is.na(affect_point) == 1){ # when i = length(all_jumps_index)
        affect_point <- length(bond_price) + 1
    }
    a_path[all_jumps_index[i]:(affect_point-1)] <- a_path[all_jumps_index[i]:(affect_point-1)] + root
}

theoretical_mod <- exp(-av_sonia * duration) * colSums(t(dpois(t(matrix(0:n_max, nrow = n_max + 1, ncol = length(duration))), lambda * duration)) * pgamma(t(matrix(threshold - a_path, ncol = n_max + 1, nrow = length(a_path))), shape = 0:n_max, rate = eta))

plot(date, bond_price, type="l", lty=1, ylim=c(0.97, 1), ann=F)
par(new=T)
plot(date, theoretical_mod, type="l", lty=2, ylim=c(0.97, 1), ann=F)
legend("topleft", legend = c("data", "estimate"), lty = 1:2)

```

```r
plot(date, a_path, type="l", lty=2, ylim=c(-5, 60), ann=F)
par(new=T)
plot(date, a_path_raw, type="l", lty=2, ylim=c(-5, 60), ann=F)
legend("topleft", legend = c("mod", "raw"), lty = 1:2)

# ratchet type valuation

price_ratchet=function(face_value, r, lambda, eta, T, dt, n_sample, threshold){
    res_list <- list()
    times <- seq(0, T, by = dt)
    price <- numeric(n_sample)
    default <- numeric(n_sample)
    n_jump <- rpois(n_sample, lambda*T)
    for(i in seq_along(n_jump)){
        if(n_jump[i] == 0){
            price[1] = face_value
        }else{
            point <- runif(n_jump[i], 0, T)
            jump <- rexp(n_jump[i], rate = eta)
            index <- numeric()
            for(j in seq_along(point)){
                index[j] <- min(which(times < point[j]))
            }
            terminal <- sum(jump)
            peak <- max(jump)
            peak_time <- index[which(jump == peak)]
            if(terminal < threshold && peak < threshold/4){
                price[i] <- face_value
            }else if(terminal < threshold && peak >= threshold/4){
                price[i] <- face_value/2
            }else {
                price[i] <- 0
            }
            res_list <- c(res_list, list(list(point, jump, index, peak_time)))
        }
    }
    discount_price <- exp(-r*T)*price
    evaluated <-mean(discount_price)
    se <- sd(discount_price)/sqrt(n_sample)
    return(list(evaluated, se, discount_price, res_list))
}


thresholds <- c(100, 200, 300, 400, 500)
years <- c(3, 4, 5)
face_value <- 1
r <- 0.001
n_sample <- 10^5
res <- price_ratchet(face_value, r, lambda, eta, 5, dt, n_sample, 100)

results <- list()

for(i in seq_along(thresholds)){
    for(j in seq_along(years)){
        results <- c(results, list(price_ratchet(face_value, r, lambda, eta, years[j], dt, n_sample,
          thresholds[i])))
    }
}
```

References

Aıt-Sahalia, Yacine (2004). “Disentangling diffusion from jumps”. In: *Journal of financial economics* 74.3, pp. 487–528.

Artemis (n.d.). *Deal Directory: MedQuake Ltd.* `https://www.artemis.bm/deal-directory/medquake-ltd/`.

Barndorff-Nielsen, Ole E and Neil Shephard (2003). “Realized power variation and stochastic volatility models”. In: *Bernoulli* 9.2, pp. 243–265.

Burnecki, Krzysztof (2005). “Pricing catastrophe bonds in a compound non-homogeneous Poisson model with left-truncated loss distributions”. In: *Mathematics in Finance Conference*.

Capiński, M. and T. Zastawniak (2016). *Credit Risk*. Mastering Mathematical Finance. Cambridge University Press. ISBN: 9781107002760. URL: https://books.google.co.jp/books?id=XAVQDQAAQBAJ.

Çınlar, E. (2011). *Probability and stochastics*. Vol. 261. Springer Science & Business Media.

Environmental Information, National Centers for (n.d.). *Search Earthquake Events*. https://www.ngdc.noaa.gov/hazel/view/hazards/earthquake/search.

Halsted, D.J. and D.E. Brown (1972). “Zakian’s technique for inverting Laplace transforms”. In: *The Chemical Engineering Journal* 3. An International Journal of Research and Development, pp. 312–313. ISSN: 0300-9467. DOI: https://doi.org/10.1016/0300-9467(72)85037-8. URL: https://www.sciencedirect.com/science/article/pii/0300946772850378.

Härdle, Wolfgang Karl and Brenda López Cabrera (2010). “Calibrating CAT bonds for Mexican earthquakes”. In: *Journal of Risk and Insurance* 77.3, pp. 625–650.

Jarrow, Robert A (2010). “A simple robust model for CAT bond valuation”. In: *Finance Research Letters* 7.2, pp. 72–79.

Kou, Steven G and Hui Wang (2004). “Option pricing under a double exponential jump diffusion model”. In: *Management science* 50.9, pp. 1178–1192.

Lepingle, D. (1976). “La variation d’ordre p des semi-martingales”. In: *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete* 36.4, pp. 295–316. DOI: 10.1007/bf00532696. URL: https://doi.org/10.1007/bf00532696.

Ma, Zong-Gang and Chao-Qun Ma (2013). “Pricing catastrophe risk bonds: A mixed approximation method”. In: *Insurance: Mathematics and Economics* 52.2, pp. 243–254.

Nowak, Piotr and Maciej Romaniuk (2013). “Pricing and simulations of catastrophe bonds”. In: *Insurance: Mathematics and Economics* 52.1, pp. 18–28.

Ramezani, Cyrus A and Yong Zeng (2007). “Maximum likelihood estimation of the double exponential jump-diffusion process”. In: *Annals of Finance* 3.4, pp. 487–507.

Utsu, Tokuji (2002). “A list of deadly earthquakes in the world: 1500–2000”. In: *International Geophysics*. Vol. 81. Elsevier, 691–cp1.

Vaugirard, Victor E (2003). “Pricing catastrophe bonds by an arbitrage approach”. In: *The Quarterly Review of Economics and Finance* 43.1, pp. 119–132.

Wei, Longfei, Lu Liu, and Jialong Hou (2022). “Pricing hybrid-triggered catastrophe bonds based on copula-EVT model”. In: *Quantitative Finance and Economics* 6.2, pp. 223–243.

近江崇宏 and 野村俊一 (2019). **点過程の時系列解析**. 統計学 one point / 鎌倉稔成 [ほか] 編 14. 共立出版. URL: https://ci.nii.ac.jp/ncid/BB28340640.

第 4 章

Occupation Time の動学的費用最小化問題への応用

三田 光星

4.1 Introduction

この論文ではある水準 y 以下で確率過程 X が過ごす時間である Occupation Time とその水準を最後に脱出した時刻である Last Exit Time の同時分布を斉時拡散過程について密度関数のラプラス変換の形で導出する．また導出された同時分布を用いて，不況を脱出し好況になるまでの政府の費用最小化問題へ応用することを考える．景気判断指標 X が景気判断基準 y より下で過ごす時間である Occupation Time は不況期の程度を測る指標として考えることができる．というのも，景気判断指標 X が y 以下の値をとることは経済の状況が悪い状態にあることを意味し，その時間が長ければ長いほど景気が芳しくないと考えられるからである．一方で Last Exit Time は景気判断指標が基準を最後に脱出する時刻であるので，経済の状況が変化し十分に安定しはじめ，好況に向かう点だと考えることができる．この 2 つの指標は水準 y に依存し，上方の点 b において確率過程 X が Kill される場合については，水準 y が高くなればどちらも大きくなることが明らかである．

Occupation Time に関する先行研究は多く存在する．レヴィ過程が 0 以上の領域で過ごす Occupation Time の分布については Bertoin (1996) 等で一般化逆正弦分布となることが指摘されている．また一般のレヴィ過程について，ある水準以下で過ごす Occupation Time が経過時間の $\alpha\%$ になるための水準 "α-quantile" y の分布は Dassios (2005) が特定している．Last Exit Time に関しては標準マルコフ過程については Getoor and Sharpe (1973) 等で，Killing Boundary を持つ拡散過程の Last Exit Time については Egami and Kevkhishvili (2017) で指摘されている．

この論文で取り扱う問題は政府が不況から好況に転じさせるための費用最小化である．問題の設定のために $t = 0$ において経済状況は不況であるとし，政府は経済を活気づけるために介入を行う．政府は景気判断指標 X を基準に経済状況を判断するとし，X が水準 b に到達した際に好況に転じたと認識する．このときの b を X の Killing Boundary とみなす．b については過去の経済データの分析等によって外部要因から設定され便宜上 kill されるまでは変更されないものと考える．すなわち，時間を通じて b は一定としておく．実際，短期において一国経済の構造の大幅な変化が何度も起こることは考えにくく，この仮定は現実に即していることが多いと言える．つまり水準 b を初めて通過する First Passage Time T_b において政府は経済政策を好況期の方針に転換する．また，簡単化のために国は財政破綻をしないものとする．

当論文は以下のような構成である. 2 章では 3 章の結果を証明するために必要な理論的背景と命題を紹介する. 命題は具体的に First Passage Time と Occupation Time の同時密度のラプラス変換 (Proposition 1), Last Exit Time に関するパスの分解式 (Proposition 2) の 2 つである. 3 章では Last Exit Time と Occupation Time の同時密度のラプラス変換を求め, X が標準ブラウン運動の場合について数値計算で逆変換を行い, 同時密度を近似する. そして最後に 4 章では費用最小化問題を定式化し, それをドリフト付きブラウン運動の場合について数値計算で解く. 付録では 2 章で省略する技術的な部分について可能な限り詳しく記載する.

4.2　数学的なフレームワーク

本章では Main Theorem の証明に必要な数学の理論的背景と命題 2 つを紹介する.

はじめに基本の設定を与える. 実現されうる事象の全体集合 Ω, Ω の σ-加法族 $\mathcal{F}$, $\mathcal{F}$ 上の確率測度 $\mathbb{P}$, usual conditions を満たすフィルトレーション $(\mathcal{F}_t)_{t\in\mathbb{R}_+}$ からなるフィルター付き確率空間 $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\in\mathbb{R}_+}, \mathbb{P})$ を考える. 正則で斉時的な拡散過程 X は $(\mathcal{F}_t)_{t\in\mathbb{R}_+}$ に適合していて 0 時では x にいるとし, X の状態空間を $\mathcal{S} = (l, r) \subset \mathbb{R}$ で l, r は natural boundary, 任意の $w \in \mathcal{S}$ について $\mathbb{P}_x(T_w < \infty) = 1$, X の Scale Function を $S(x)$, Speed Measure を m で表し, m に対する X の推移確率密度を $p(t, x, y)$ としておく. X の無限小パラメーター $\mu(x)$ と $\sigma^2(x)$ について作用素 $\mathcal{L}_X$ を

$$\mathcal{L}_X = \mu(x)\frac{\partial}{\partial x} + \frac{1}{2}\sigma^2(x)\frac{\partial^2}{\partial x^2}$$

で定義する. また T_z を z の First Passage Time とすると, $(x-y)(x-z) < 0$ となる x, y, z について

$$\mathbb{P}_x(T_z < T_y) = \frac{S(x) - S(y)}{S(z) - S(y)}$$

が成立する. b を終点とするときの Last Exit Time を $L_y \coloneqq \sup\{t < T_b \mid X_t = y\}$, L_y までの Occupation Time を $A_y \coloneqq \int_0^{L_y} \mathbb{1}_{\{X_u \le y\}} du$ とする. 通常通り, $\sup \emptyset = 0$ と考える. また X が途中で Kill されるとき, Cemetery Point $\Delta \notin \mathcal{S}$ へ移動するものとし, X の生存時間を $\xi \coloneqq \inf\{t \mid X_t = \Delta\}$ で定義すると $\mathbb{P}_x(\xi < \infty) = 1$ である状況を考える.

4.2.1　First Passage Time と Occupation Time のラプラス変換

本節では First Passage Time T_z において Kill される過程について, その生存時間である T_z と Occupation Time の同時分布について求める. 詳細は Zhang (2015) を参照のこと.

各 $\alpha > 0$ について, Sturm-Liouville 方程式 “ $\mathcal{L}_X(f) = \alpha f$ ” の解のうち単調増加のものを ϕ_α^+, 単調減少のものを ϕ_α^- とする. 加えて $w_\alpha \cdot S'(x) = (\phi_\alpha^+(x))'\phi_\alpha^-(x) - \phi_\alpha^+(x)(\phi_\alpha^-(x))'$ を満たす定数 $w_\alpha > 0$ について $W_\alpha(x, y)$ を,

$$W_\alpha(x, y) \coloneqq w_\alpha^{-1} \det \begin{bmatrix} \phi_\alpha^+(x) & \phi_\alpha^+(y) \\ \phi_\alpha^-(x) & \phi_\alpha^-(y) \end{bmatrix}$$

と定義する. また, $W_0(x, y) \coloneqq S(x) - S(y)$ と定義しておく. $W_{\alpha,1}(x, y) = \partial W_\alpha / \partial x$, T_z までの Occupation Time を $B_y^z \coloneqq \int_0^{T_z} \mathbb{1}_{\{X_u \le y\}} du$ と書くこととして, First Passage Time T_z と First Passage Time までの Occupation Time B_y^z の同時分布は以下のラプラス変換 $F_{\alpha,\beta}(x, y, z)$ の形で表現できる.

Proposition 2 (cf. Corollary 1 of Zhang (2015))**.** $x, y \in (l, z)$, $\alpha \geq 0$, $\beta > 0$ について以下が成立する.

$$\begin{aligned}
F_{\alpha,\beta}(x,y,z) &:= \mathbb{E}_x[\exp(-\beta B_y^z - \alpha T_z)] \\
&= \frac{W_\alpha(x,y)}{W_\alpha(z,y)} + \frac{(W_\alpha(z,x)/W_\alpha(z,y))S'(y)}{W_{\alpha,1}(y,z) + W_\alpha(z,y)(\phi^{+\prime}_{\beta+\alpha}(y)/\phi^{+}_{\beta+\alpha}(y))} && \text{if } x \in (y,b) \\
&= \frac{\phi^{+}_{\beta+\alpha}(x)}{\phi^{+}_{\beta+\alpha}(y)} \frac{S'(y)}{W_{\alpha,1}(y,z) + W_\alpha(z,y)(\phi^{+\prime}_{\beta+\alpha}(y)/\phi^{+}_{\beta+\alpha}(y))} && \text{if } x \in (l,y]
\end{aligned}$$

証明の概要は付録 A.1 で後述する. Sturm-Liouville 方程式 “ $\mathcal{L}_X(f) = \alpha f$ ” の単調解 $\phi^{\pm}$ は任意の固定された $\kappa \in \mathcal{S}$ についてそれぞれ

$$\phi^+_\alpha(x) = \begin{cases} \mathbb{E}_x[\mathrm{e}^{-\alpha T_\kappa}] & \text{if } x \leq \kappa \\ 1/\mathbb{E}_\kappa[\mathrm{e}^{-\alpha T_x}] & \text{if } x > \kappa \end{cases} \qquad \phi^-_\alpha(x) = \begin{cases} 1/\mathbb{E}_\kappa[\mathrm{e}^{-\alpha T_x}] & \text{if } x \leq \kappa \\ \mathbb{E}_x[\mathrm{e}^{-\alpha T_\kappa}] & \text{if } x > \kappa \end{cases}$$

のように書けることが Zhang (2015) で指摘されている. 解に登場する期待値は First Passage Time の分布のラプラス変換であるので, 関数 $F_{\alpha,\beta}$ はスケール関数と First Passage Time の分布により特定することが可能である. また $\beta \to 0$ のとき, 優収束定理と付録 A.1 の Lemma 1 から $F_{\alpha,\beta}(x,y,z) \to \mathbb{E}_x[\exp(-\alpha T_z)] = \phi^+_\alpha(x)/\phi^+_\alpha(z)$ である.

4.2.2 Last Exit Time とパスの分解

続いて Last Exit Time に関する命題について述べる. 状態空間 $\mathcal{S}$ 内の全ての $x < y$ において $\mathbb{P}_x(T_y < \infty) = 1$ が成立する拡散過程 X を考える. これは拡散過程 X が 0 時における点から上方には確率 1 で有限時間内に到達するということを意味しており, 下方の点への到達の条件は求められていない. 生存時間 $\xi > 0$ である場合, $z = \sup\{X_t \mid 0 \leq t < \xi\}$ で条件づけた X のパスの分布は, T_z で Kill されるパスの分布と T_z で Kill されるものを時間反転したパスの分布という 2 つの独立なものに分解が可能である (付録 A.2). この事実を用いて Last Exit Time まで生存するパスの分布に関する式を導出する.

x から始まる生存時間 ξ までのパスの空間上で定義される分布を $\mathbb{P}_x^\xi$ と書く. また, パスの空間上に定義される分布 Q と Q' について $Q^\wedge$ を Q の時間反転により得られる像, $Q \circ Q'$ をそれぞれの分布に従う独立なパスを順番に連結したパスの分布というように書けば, 生存時間が Last Exit Time L_y である X のパスの分布について以下が成立する.

Proposition 3. 拡散過程 X は状態空間 $\mathcal{S}$ 内の全ての $x < y$ において $\mathbb{P}_x(T_y < \infty) = 1$ であり, *Killing Boundary* b を持つとする. このとき, 任意の $x, y \in (l, b)$ について以下が成立する.

$$\mathbb{P}_x^{L_y}(\,\cdot\,,\ L_y > 0) = \frac{1}{S(b) - S(y)} \int_{x \vee y}^{b} S(dz) \mathbb{P}_x^{T_z} \circ (\mathbb{P}_y^{T_z})^\wedge(\,\cdot\,)$$

パスの分解や Last Exit Time の分布に関する詳細は付録で後述する.

4.3 Main Theorem

4.3.1 Main Theorem

前述の Proposition を使うことにより, Occupation Time と Last Exit Time の分布は以下のようなラプラス変換の形で得ることができる.

Theorem. 斉時拡散過程 X は状態空間 $\mathcal{S}$ 内の全ての $x < y$ において $\mathbb{P}_x(T_y < \infty) = 1$ が成立し, *Killing Boundary* b を持つものとする. このとき, $x, y \in (l, b)$, $\alpha \geq 0, \beta > 0$ について以下が成立する.

$$\mathbb{E}_x[\exp(-\beta A_y - \alpha L_y) \mid L_y > 0] = \frac{1}{S(b) - S(x \vee y)} \int_{x \vee y}^{b} S(dz) F_{\alpha,\beta}(x, y, z) F_{\alpha,\beta}(y, y, z)$$

Proof. $\mathbb{E}_x[\exp(-\beta A_y - \alpha L_y) \mid L_y > 0] = \frac{1}{\mathbb{P}_x(L_y > 0)} \mathbb{E}_x[\mathrm{e}^{-\beta A_y - \alpha L_y} \mathbb{1}_{\{L_y > 0\}}]$ である. Last Exit Time が 0, つまり $\{t < T_b \mid X_t = y\} = \emptyset$ となるのは y より先に b に到達してしまう場合であるので,

$$\mathbb{P}_x(L_y > 0) = \begin{cases} \frac{S(b) - S(x)}{S(b) - S(y)} & \text{if } x > y \\ 1 & \text{if } x \leq y \end{cases} \tag{4.1}$$

となる. また, 推移密度が対称であることから $(\mathbb{P}^t_{x,y})^\wedge = \mathbb{P}^t_{y,x}$ が明らかであることに注意して,

$$\begin{aligned} \mathbb{E}_x[\mathrm{e}^{-\beta A_y - \alpha L_y} \mathbb{1}_{\{L_y > 0\}}] &= \mathbb{E}_x^{L_y}[\mathrm{e}^{-\beta A_y - \alpha L_y} \mathbb{1}_{\{L_y > 0\}}] \\ &= \frac{1}{S(b) - S(y)} \int_{x \vee y}^{b} S(dz) \mathbb{E}_x^{T_z}[\mathrm{e}^{-\beta B_y^z - \alpha T_z}] \mathbb{E}_y^{T_z}[\mathrm{e}^{-\beta B_y^z - \alpha T_z}] \\ &= \frac{1}{S(b) - S(y)} \int_{x \vee y}^{b} S(dz) F_{\alpha,\beta}(x, y, z) F_{\alpha,\beta}(y, y, z) \end{aligned} \tag{4.2}$$

なお $\mathbb{E}_x^\xi$ は生存時間を ξ とするパス空間上の測度での積分を表す. Proposition 1 は First Passage Time までについての命題であるので最後の等式が成立する. (4.1), (4.2) を合わせれば定理の式が得られる. □

4.3.2 ラプラス変換と同時密度関数の例

Occupation Time と Last Exit Time の同時密度は Theorem 1 で求めた関数に対して逆ラプラス変換を施すことで得られる. ここでは標準ブラウン運動 $X_t = B_t$ を例に, Quadrature Method で逆ラプラス変換を数値的に行った. 詳細は付録 B のスクリプトを参照すること. ブラウン運動について Zhang (2015), Borodin and Paavo Salminen (2015) により, Scale Function を $S(x) = x$ で固定すれば

$$\begin{aligned} &S(dz) = dz, \quad \phi_\alpha^\pm(x) = \mathrm{e}^{\pm\sqrt{2\alpha}x}, \\ &W_\alpha(x, y) = \frac{1}{2\sqrt{2\alpha}} \sinh(\sqrt{2\alpha}(x - y)), \quad W_0(x, y) = x - y \\ &W_{\alpha,1}(x, y) = \frac{1}{2} \cosh(\sqrt{2\alpha}(x - y)), \quad W_{0,1}(x, y) = 1 \end{aligned}$$

以上を用いて Theorem を適用する. 4 章での費用最小化問題の設定において $x \le y$ を仮定するので, それにならい $x \le y$ の場合について考えることにすれば, $\gamma_\pm = \sqrt{2(\alpha+\beta)} \pm \sqrt{2\alpha}$ として $\alpha > 0$ のとき

$$\begin{aligned}
\mathbb{E}_x[\mathrm{e}^{-\beta A_y - \alpha L_y}] &= \frac{1}{b-y}\int_y^b dz\, \mathrm{e}^{\sqrt{2(\alpha+\beta)}(x-y)} \\
&\qquad \times \left(\frac{2\sqrt{2\alpha}}{\sqrt{2\alpha}\cosh(\sqrt{2\alpha}(z-y)) + \sqrt{2(\alpha+\beta)}\sinh(z-y)}\right)^2 \\
&= \frac{1}{\gamma_+(b-y)}\left[1 - \frac{2\sqrt{2\alpha}}{\gamma_+ \exp(2\sqrt{2\alpha}(b-y)) - \gamma_-}\right]\mathrm{e}^{\sqrt{2(\alpha+\beta)}(x-y)} \qquad (4.3)
\end{aligned}$$

であり, $\alpha = 0$ のとき

$$\begin{aligned}
\mathbb{E}_x[\exp(-\beta A_y)] &= \frac{1}{b-y}\int_y^b \mathrm{e}^{\sqrt{2\beta}(x-y)}\left[\frac{1}{1+\sqrt{2\beta}(z-y)}\right]^2 dz \\
&= \frac{1}{\sqrt{2\beta}(b-y)}\left[1 - \frac{1}{1+\sqrt{2\beta}(b-y)}\right]\mathrm{e}^{\sqrt{2\beta}(x-y)}
\end{aligned}$$

となる. $\beta \to 0$ において Last Exit Time L_y のラプラス変換と一致することより必要条件を満たしていることは容易に確認できる. この式に $x = 0, y = 3, b = 10$ を代入して逆ラプラス変換を行なった. そうして導出された 3 次元のグラフが図 1, $l \times 10^3 = 100, 500, 700$ の平面で切ったグラフが図 2 である. なお, 計算の過程で Occupation Time a が Last Exit Time l より大きくなる領域においては 0 となるように設定してあるため, $a > l$ の領域は 0 となっている. 実際, 定義より Occupation Time が Last Exit Time より大きくなることはない. 加えて数値計算上の特異点の影響で負になる場所についても 0 を代入している.

4.3.3 Last Exit Time の条件付き分布

Last Exit Time は $\mathcal{F}_t$ について可測ではなく, Stopping Time ではない. つまり実際に確率過程が kill されるまでは y 以下の領域を脱出したかどうかはわからないといった性質を持つ変数である. 一方で Occupation Time は adapted な確率過程であるため, 上で導出した同時分布を用いることで Last Exit Time の条件付き分布を知ることが可能となる. 具体的に同時密度を $f(a, l)$ で表すことにすると, 条件付き密度の定義より

$$\mathbb{P}(L_y \in dl \mid A_y = a) = f(l \mid a)dl = \frac{f(a,l)}{f(a)}dl$$

と計算できる. Occupation Time の分布はたとえば Karatzas et al. (1991) の 3.6 節などのように Local Time と関連して議論されているほか, 標準ブラウン運動における $(0, \infty)$ の Occupation Time の分布は

$$\begin{aligned}
\mathbb{P}\left(\int_0^t \mathbb{1}_{(0,\infty)}(W_s)ds \le \theta\right) &= \int_0^{\theta/t} \frac{ds}{\pi\sqrt{s(1-s)}} \\
&= \frac{2}{\pi}\arcsin\sqrt{\frac{\theta}{t}} \qquad (4.4)
\end{aligned}$$

という形であることが P. Lévy's Arcsine Law という結果として知られているなど, 十分に研究されている加法的汎関数であるため, 条件付き密度は十分導出可能な関数であると考えられる. ここでブラウン運動は対称であるため, $(-\infty, 0)$ と同じであることに注意する.

Last Exit Time の条件付き分布を知ることは非常に有益である. たとえば確率過程 X を日経平均, y を不況, b を好況の水準とし, y 以下の領域を脱出した時刻 τ_y が Last Exit Time であれば τ_y で不況を脱したと考えるこ

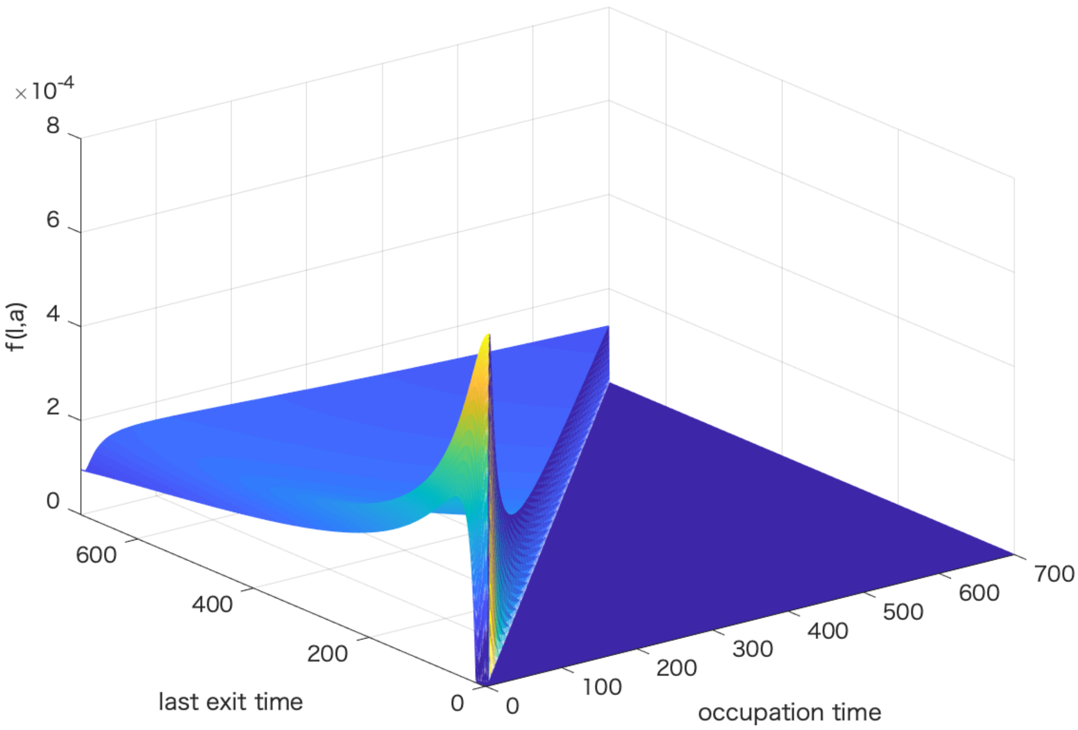

図 4.1　Last Exit Time と Occupation Time の同時密度関数. x, y 軸ともにスケールは 10^{-3}

ととする．この状況においては τ_y が Last Exit Time である確率が不況を脱して好況に転換する確率となるが, Occupation Time の情報を使わない $\mathbb{P}(L_y \in dl)$ でその確率を推計するよりも, Occupation Time を使った条件付き密度 $\mathbb{P}(L_y \mid A_y)$ の方が考慮する事象が限定されており, 推計の精度が上がると考えられ, 不況における財政出動を取りやめるなどの判断がより正確に可能になる．したがって条件付き密度を考えることは非常に有意義であるといえる．

ここでは具体例として X が標準ブラウン運動で $y = 0$ の場合を考える．同時密度 $f(a, l)$ のラプラス変換は (4.3) で表されることから,

$$\mathbb{E}_x[\mathrm{e}^{-\beta A_0 - \alpha L_0}] = \frac{1}{\gamma_+ b}\left[1 - \frac{2\sqrt{2\alpha}}{\gamma_+ \exp(2b\sqrt{2\alpha}) - \gamma_-}\right]\mathrm{e}^{\sqrt{2(\alpha+\beta)}x}$$

である．これを数値的に逆ラプラス変換した同時密度を $\hat{f}(a, l)$ とする．また, P. Lévy's Arcsine Law より Occupation Time の密度は (4.4) で表されることと arcsin の微分より, 時刻 t における条件付き密度は

$$\begin{aligned}\hat{f}(l \mid a) &= \frac{\hat{f}(a, l)}{f(a)} \\ &= \hat{f}(a, l)\pi\sqrt{a(t-a)}\end{aligned}$$

と計算できる．

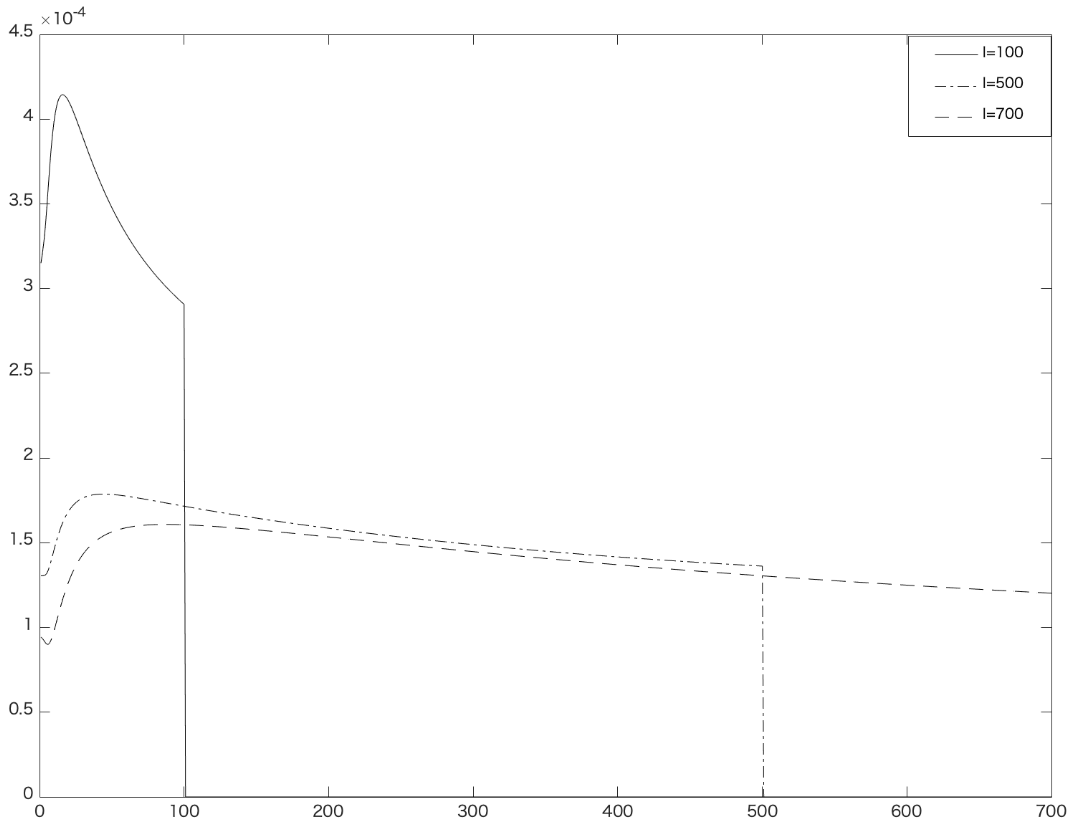

図 4.2　$L_y = l$ のときの Occupation Time の密度. 横軸のスケールは 10^{-3}

4.4　費用最小化問題への応用

この章では 3 章で求めた Occupation Time と Last Exit Time の同時分布を費用最小化問題へ応用する. 当論文で考える費用最小化問題の設定は以下の通りである.

政府はまず景気判断水準 y を設定する. その水準 y を下回る時間は不況であることから失業者に対する社会保障費等, 国民の生活支援を行うこととする. 対して, 水準 y を最後に脱出した時から好況になったとみなす水準の b に到達するまでの間は景気回復の後押しに資金を投入することとする.

このような設定においては景気判断水準 y を低く設定することで不況であると判断する時間, つまり Occupation Time が短くなり, 社会保障費などの費用が下がる. その一方で最後に不況の水準 y を脱出してから水準 b に到達するまでの時間は長くなってしまい, 景気回復の後押しに投入する資金が余分にかかってしまう. このようなトレードオフ関係にある 2 つの費用を最小化する最適な y を設定することが今回の目標となる. また今回は不況からの脱出を考えるという設定上, 0 時における景気判断指標の値 x より下に y を設定できないようにしておく. つまり $x \leq y$ である.

4.4.1 定式化

上記設定において, 費用は Occupation Time A_y と Last Exit Time L_y の 2 つを引数として考えることができる. というのも不況脱出から好況になったと判断するまでの時間 $T_b - L_y$ に関して操作できる変数が y のみの場合, y に依存しているのは L_y のみだからである. L_y だけでなく T_b も確率変数であり $\omega \in \Omega$ に依存するので, 扱う引数を $T_b - L_y$ ではなく L_y として ω に依存する変数を減らすことはモデルを単純にするという意味で有用であると考えられる. また $T_b - L_y$ の分布について Egami and Kevkhishvili (2017) でドリフト付ブラウン運動については特定されているが, 一般の拡散過程については求められていない. また, ドリフト付ブラウン運動で特定されている分布もラプラス変換の形でしか導出されていない密度と極限を用いた形となっており扱いが非常に難しい. よって $T_b - L_y$ をそのままモデルに導入する場合はシミュレーションによる解法が考えられる. ここでドリフト付きブラウン運動 $X_t = t + B_t$ について $b = 10$ として 10 万回シミュレーションを行い $T_b - L_y$ と L_y をグラフにしたものがそれぞれ図 3, 図 4 である. 図から明らかに 10 万回のシミュレーションでも両者

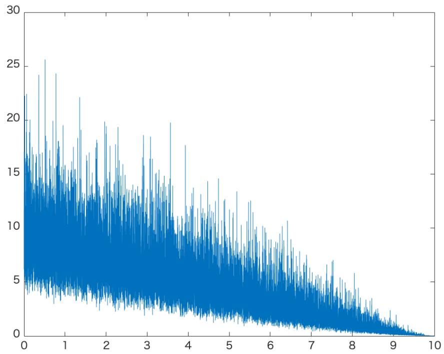

図 4.3　$T_b - L_y$ のグラフ

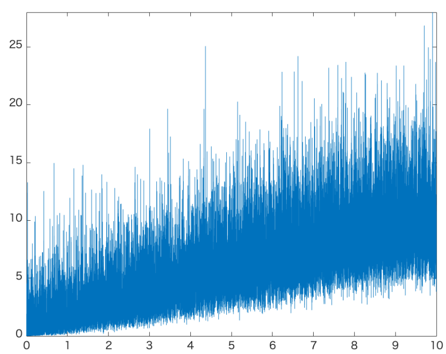

図 4.4　L_y のグラフ

ともに分散が非常に大きく, シミュレーションで上記の引数をとる費用最小化問題を解くことは推奨されない. よってシミュレーションするほかない $T_b - L_y$ ではなく, Theorem で分布が導出できた L_y を使うことは合理的であると考えられる. ここで $C(a, l\,|\,y)$ を a に関して増加, l に関して減少で十分に滑らかな連続関数とする. 定義と X の連続性より明らかに A_y と L_y は y に関して増加であるので, 当該費用最小化問題における費用関数は $C(A_y, L_y\,|\,y)$ の形で書ける. よってこの費用最小化問題は

$$\min_{y \in (l,b)} \mathbb{E}_x[C(A_y, L_y\,|\,y)]$$

と定式化できる.

4.4.2 具体例

$(X_t)_{t \in \mathbb{R}_+}$ をドリフト付きブラウン運動, 費用関数を $C(a, l) = 1 - \mathrm{e}^{-\beta a}(1 - e^{-\alpha l})$ と置く. この費用関数の第 2 項は a が減少すれば増加する $\mathrm{e}^{-\beta a}$ と l が増加すれば増加する $(1 - e^{-\alpha l})$ の積で表されている. それぞれ A_y と $T_b - L_y$ に関してどの程度小さくできたかというスコアを表しており, C はそのスコアが大きければ大きいほど

費用が小さくなるという関数である. この費用関数について費用最小化問題は

$$\min_{y\in(l,b)} \mathbb{E}_x[1-\mathrm{e}^{-\beta A_y}(1-e^{-\alpha L_y})] = \min_{y\in(l,b)}(1-\mathbb{E}_x[\mathrm{e}^{-\beta A_y}]+\mathbb{E}_x[e^{-\beta A_y-\alpha L_y}])$$

という形でかける. Theorem を用いれば目的関数は

$$\begin{aligned}1-\mathbb{E}_x[\mathrm{e}^{-\beta A_y}]+\mathbb{E}_x[e^{-\beta A_y-\alpha L_y}] &= 1-\frac{1}{S(b)-S(y)}\int_y^b S(dz)F_{0,\beta}(x,y,z)F_{0,\beta}(y,y,z)\\ &\quad +\frac{1}{S(b)-S(y)}\int_y^b S(dz)F_{\alpha,\beta}(x,y,z)F_{\alpha,\beta}(y,y,z)\end{aligned}$$

となる. ここで Zhang (2015), Borodin and Paavo Salminen (2015) より $\delta := \mu/\sigma^2$, $\gamma_\alpha := \sqrt{\delta^2+2\alpha/\sigma^2}$ として Scale Function を $S(x) = (1-\exp(-2\delta x))/\delta$ と固定しておくと,

$$\begin{aligned}&S(dz) = 2\mathrm{e}^{-2\delta x}dz, \quad \phi_\alpha^+(x) = \mathrm{e}^{(\gamma_\alpha-\delta)x}, \quad \phi_\alpha^-(x) = \mathrm{e}^{-(\gamma_\alpha+\delta)x}\\ &W_\alpha(x,y) = 2\mathrm{e}^{-\delta(x+y)}\frac{\sinh(\gamma_\alpha(x-y))}{\gamma_\alpha}\\ &W_{\alpha,1}(x,y) = -2\delta\mathrm{e}^{-\delta(x+y)}\frac{\sinh(\gamma_\alpha(x-y))}{\gamma_\alpha}+2\mathrm{e}^{-\delta(x+y)}\cosh(\gamma_\alpha(x-y))\end{aligned}$$

である. 上式を見ると明らかだが, ドリフト付きブラウン運動は基本的に各パラメーターごとの比率によって形が決まる. よって $\sigma = 3$, $x = 0$, $b = 50$, $\alpha = \beta = 0.05$ を固定する. μ を 0.5 から 2.5 まで 1 ずつ変化させた y と費用関数のグラフが図 5 である.

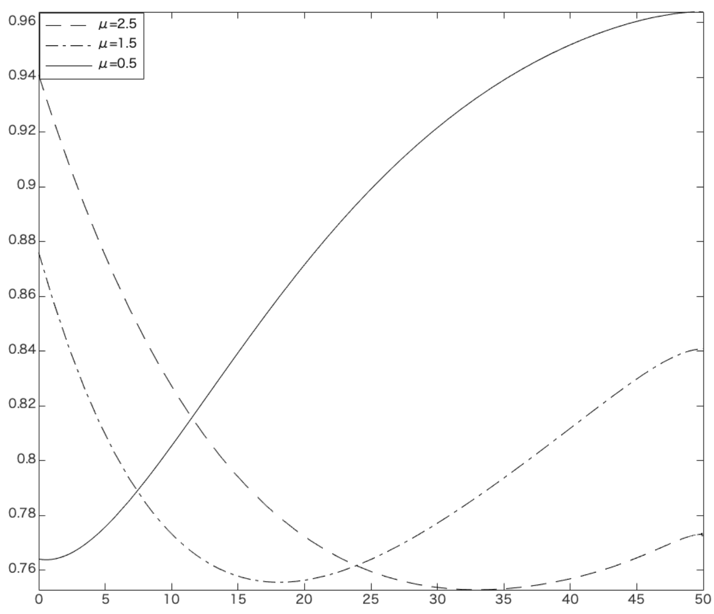

図 4.5　費用関数と水準 y に関するグラフ. 左上から $\mu = 2.5, 1.5, 0.5$

図を見ると明らかなようにドリフトが大きくなるほど最適な y も大きくなる. 大きいドリフトは景気回復が速いことを示唆するので, この場合は景気回復後押しのコストを大きく減らすことが優先される. というのは,

少々高い水準を設定していても高い回復力ですぐに平常時に戻ってこられるため, Occupation Time 自体がかなり小さくなることが期待されるからである. この事実は直観に反していない.

4.5 結論

Last Exit Time とそれまでの Occupation Time の分布のラプラス変換を, 上方の点への到達確率が 1 となる拡散過程のパスの分解を用いることで積分の形で求めることができ, さらに標準ブラウン運動についてはその積分を明示的に書くことができた. ラプラス変換は一意でありこれを逆変換することで Occupation Time と Last Exit Time の密度関数が得られるため, その密度関数を使ってこの 2 つの指標についてさまざまな応用が可能となる. ただし当論文では密度関数の形を特定, あるいは推定するまでには至っておらず, 結果として費用最小化問題への応用においても費用関数の形が制限されている. 今後の課題としては数値計算によって可能な限り正確に密度関数を導出し, 一般の関数についても適用できるようにしていくことが考えられる.

なお, ラプラス変換の形でしか明示的に表せていない関数は多く存在する. この論文で示した式は状態空間上の上方の点に確率 1 で有限時間で到達する拡散過程すべてについて成立するので, ブラウン運動に限らずさまざまな拡散過程について適用できることは重要な点である.

また, 当論文では簡単化のために財政破綻の可能性を排除し必ず好況になる場合について考えたが, 実際はひどい不況の結果財政破綻を起こす場合もある. その場合についても下方向の Killing Boundary a を設定し, Zhang (2015) の Proposition 3 を用いれば同様の議論を行うことができる. このとき Theorem の条件付期待値には倒産するか上場できるかが追加され, 上場できる場合は $\mathbb{E}_x[\mathrm{e}^{-\beta A_y - \alpha L_y} \mid L_y > 0,\ T_b < T_a]$ というように書かれる.

最後に 4.3.3 節でも確認した通り, Last Exit Time L_y は X から生成されるフィルトレーションに対して Stopping Time ではないが, y より下を抜け出した時にその点が Last Exit Time である条件付き確率は Theorem の式に逆ラプラス変換を施すことで理論上計算できる. ただし現状では同時密度関数の導出には至っていない上, 仮に導出が可能であったとしてもその点が実際に Last Exit Time であるかは b に到達するまでは確率の値でしか判断ができない. つまり景気の後押しを行う判断のタイミングは為政者の意思決定に依存し, この費用最小化問題は最適停止問題の要素も含むこととなる. 仮に意思決定のタイミングが Last Exit でない場合に, 景気後押しの資金に加えて y を下回ってしまう場合の社会保障費も追加でかかることや, Last Exit Time で景気の後押しに資金を投入するという意思決定を行わずに b に達してしまった場合のペナルティ (例: 政策の失敗による支持率の低下の信用回復費用など) も問題に組み込むべきである.

付録 4.A　各 Proposition の証明

この節では本文で省略した内容について書くこととする.

4.A.1　Proposition 1 の証明

Proposition 1 の証明に入る前に必要な Lemma を 2 つ紹介する. これら Lemma の証明等, 詳細は Zhang (2015) を参照のこと.

Lemma 1. $\mathcal{S}$ を X の状態空間とし, $x, y, z \in I$ が $(x-y)(z-x) > 0$ を満たすとする. このとき $\alpha \geq 0$, $e_\alpha \sim \exp(\alpha)$ について以下を得る.

$$\mathbb{E}_x[\mathrm{e}^{-\alpha T_y} \mid T_y < T_z] = \mathbb{P}_x(T_y < T_z \wedge e_\alpha) = \frac{W_\alpha(x,z)}{W_\alpha(y,z)}$$

また $|z| \to \infty$ とすると, $\mathbb{E}_x[\mathrm{e}^{-\alpha T_y}] = \phi_\alpha^+(x)/\phi_\alpha^+(y)$ が成立する.

Lemma 2. $x, y, z \in \mathcal{S}$ と $\alpha > 0$ について以下が成立する.

$$W_\alpha(x,y) = -W_\alpha(y,x)$$
$$\frac{\partial}{\partial x}\frac{W_\alpha(x,y)}{W_\alpha(x,z)} = \frac{S'(x)}{W_\alpha^2(x,z)}W_\alpha(y,z)$$

Proposition 1 の証明. 強マルコフ性と Lemma 1 を用いると以下のように分解できる.

$$\begin{aligned}\mathbb{E}_x[\exp(-\beta B_y^z - \alpha T_z)] &= \mathbb{E}_x[\mathrm{e}^{-\alpha T_z} \mid T_z < T_y] + \mathbb{E}_x[\exp(-\beta B_y^z - \alpha T_z) \mid T_y < T_z] \\ &= \begin{cases} \frac{\phi_\alpha^+(x)}{\phi_\alpha^+(y)}\mathbb{E}_y[\exp(-\beta B_y^z - \alpha T_z)] & x \leq y \\ \frac{W_\alpha(x,y)}{W_\alpha(z,y)} + \frac{W_\alpha(x,z)}{W_\alpha(y,z)}\mathbb{E}_y[\exp(-\beta B_y^z - \alpha T_z)] & x > y. \end{cases}\end{aligned}$$

よって $\mathbb{E}_y[\exp(-\beta B_y^z - \alpha T_z)]$ について考えれば十分である. $\epsilon > 0$ を $y + \epsilon < b$ となるようにとり, $B_{y,\epsilon}^z$ を B_y^z の近似として以下のように定義する (図 6).

$$B_{y,\epsilon}^z = \sum_{n=1}^{\infty} T_{y+\epsilon}^{+,n} \wedge T_z - T_y^{-,n} \wedge T_z$$
$$T_{y+\epsilon}^{+,n} := \inf\{t \geq T_y^{-,n} \mid X_t \geq y+\epsilon\} \quad T_y^{-,n+1} := \sup\{t \geq T_{y+\epsilon}^{+,n} \mid X_t \leq y\} \quad n \geq 1$$

ただし $T_y^{-,1} = T_y$ であるとする. $B_{y,\epsilon}^z$ は図 6 の $T_y^{-,2} - T_{y+\epsilon}^{+,1}$ のような上から下に抜けるまでの時間は含まないことに注意して, 強マルコフ性と X の連続性から,

$$\begin{aligned}\mathbb{E}_y[\exp(-\beta B_{y,\epsilon}^z - \alpha T_z)] &= \mathbb{E}_y[\exp(-(\beta+\alpha)T_{y+\epsilon})]\mathbb{E}_{y+\epsilon}[\exp(-\beta B_{y,\epsilon}^z - \alpha T_z)] \\ &= \mathbb{E}_y[\exp(-(\beta+\alpha)T_{y+\epsilon})](\mathbb{E}_{y+\epsilon}[\exp(-\alpha T_z) \mid T_z < T_y] \\ &\quad + \mathbb{E}_{y+\epsilon}[\exp(-\alpha T_y) \mid T_y < T_z]\mathbb{E}_y[\exp(-\beta B_{y,\epsilon}^z - \alpha T_z)]).\end{aligned}$$

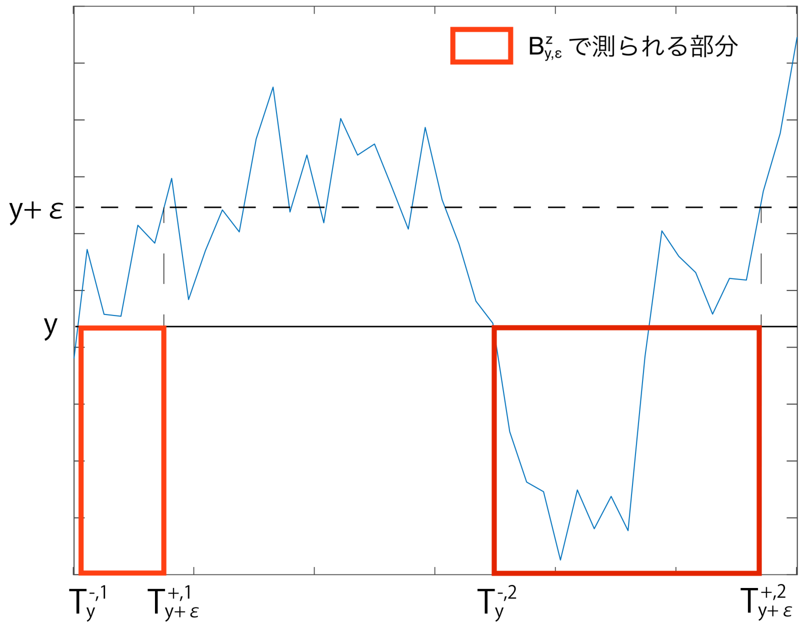

図 4.6　$T^{-,n}_y$ と $T^{+,n}_{y+\epsilon}$ の関係について

であるので, $\mathbb{E}_y[\exp(-\beta B^z_{y,\epsilon} - \alpha T_z)]$ について式を整理すると

$$
\begin{aligned}
\mathbb{E}_y[\exp(-\beta B^z_{y,\epsilon} - \alpha T_z)] &= \frac{W_\alpha(y+\epsilon, y)}{W_\alpha(z,y)} \frac{\phi^+_{\beta+\alpha}(y)/\phi^+_{\beta+\alpha}(y+\epsilon)}{1-(\phi^+_{\beta+\alpha}(y)/\phi^+_{\beta+\alpha}(y+\epsilon))(W_\alpha(y+\epsilon,z)/W_\alpha(y,z))} \\
&= \left\{\frac{W_\alpha(y+\epsilon,y)}{W_\alpha(y+\epsilon,z)}\right\} \\
&\quad \times \frac{-1}{(W_\alpha(y,z)/\phi^+_{\beta+\alpha}(y))(\phi^+_{\beta+\alpha}(y+\epsilon)/W_\alpha(y+\epsilon,z) - \phi^+_{\beta+\alpha}(y)/W_\alpha(y,z))}
\end{aligned}
$$

$B^z_{y,\epsilon}$ は X が y 以下で過ごす時間と y から $y+\epsilon$ へ移動している時間の和であるが, $\epsilon \to 0$ とすると後者が $\mathbb{P}_x$ について $a.s.$ で 0 に収束するため, $B^z_{y,\epsilon} \to B^z_y$ である. また, $0 < \exp(-\beta B^z_{y,\epsilon} - \alpha T_z) \leq 1$ より優収束定理から

$$
\begin{aligned}
\mathbb{E}_y[\exp(-\beta B^z_y - \alpha T_z)] &= \mathbb{E}_y\left[\lim_{\epsilon\to 0} \exp(-\beta B^z_{y,\epsilon} - \alpha T_z)\right] \\
&= \lim_{\epsilon\to 0} \mathbb{E}_y[\exp(-\beta B^z_{y,\epsilon} - \alpha T_z)] \\
&= \lim_{\epsilon\to 0}\left[\left\{\frac{W_\alpha(y+\epsilon,y)}{W_\alpha(y+\epsilon,z)}\right\}\right. \\
&\quad \left.\times \frac{-1}{(W_\alpha(y,z)/\phi^+_{\beta+\alpha}(y))(\phi^+_{\beta+\alpha}(y+\epsilon)/W_\alpha(y+\epsilon,z) - \phi^+_{\beta+\alpha}(y)/W_\alpha(y,z))}\right] \\
&= \frac{S'(y)}{W_{\alpha,1}(y,z) + W_\alpha(z,y)(\phi^{+\prime}_{\beta+\alpha}(y)/\phi^+_{\beta+\alpha}(y))}
\end{aligned}
$$

以上より示せた. □

4.A.2 Proposition 2 の証明

Proposition 2 の証明のために 2 つの補題を紹介する. 1 つは状態空間 $\mathcal{S} = [0, \infty)$ の拡散過程の橋についての調和式である.

Lemma 3 (Corollary 3 of Pitman and Yor (1996))**.** *$x, y \in \mathcal{S}$ で $x < y$ を満たすとき, $\mathbb{P}_x(T_y < \infty) = 1$ であると仮定する. このとき, すべての $x, y \in \mathcal{S}$ について以下のパスの空間上の測度に関する調和式が成立する.*

$$\int_0^\infty dt p(t, x, y)\mathbb{P}^t_{x,y} = \int_{x\vee y}^\infty S(dz)(\mathbb{P}^{T_z}_x) \circ (\mathbb{P}^{T_z}_y)^\wedge$$

Lemma 1 は Excursion Theory における伊藤の Excurison Law の二つの表現法が同値であることを意味する式である.

(左辺): ある t を推移密度 $p(t, x, y)dt$ によって選び, 長さ t の X の橋を x から y まで走らせる

(右辺): z を $[x \vee y, \infty)$ に制限して Speed measure $S(dz)$ によって選び, X を x から T_z まで走らせたものと y から初めて T_z まで走らせたものを連結する

つまり (左辺) は生存時間で条件付けた場合についての式であり, (右辺) は最大値で条件付けた場合についての式となっている. 詳しくは Rogers and Williams (2000), Pitman and Yor (1982) 等を参照すること. Pitman and Yor (1996) では状態空間を $\mathcal{S} = [0, \infty)$ としているが, 実際は条件を満たす過程において空でなければ問題がない. その場合, z は $[x \vee y, r)$ に制限して積分すればよい.

もう 1 つの補題は Last Exit Time の密度についてである.

Lemma 4 (Proposition 3.1 of Egami and Kevkhishvili (2017))**.** *x, $y \in (l, b)$ であるとする. b を Killing Boundary として持つ拡散過程 X について, Last Exit Time の密度は speed measure m に対する推移密度 $p(t, x, y)$ を用いて以下のように表せる.*

$$\mathbb{P}_x(L_y \in dt, L_y > 0) = \frac{p(t, x, y)}{S(b) - S(y)}dt.$$

証明の概要は以下の通りである. minimal excessive function h によって $T_y < T_b$ かつ生存時間が y の Last Exit Time となるように条件付けた拡散過程 X と同じ Law を持つ過程 X^h を生成すると, その X^h の生存時間 ξ が元の過程 X の Last Exit Time と同じ分布を持つ. その ξ の分布を Martin 関数を用いて特定する. 詳しい議論については Egami and Kevkhishvili (2017), Salminen (1984) を参照すること.

Proof of Proposition 2. $\mathbb{P}^{L_y}_x(\cdot \mid L_y = t) = \mathbb{P}^t_x(\cdot \mid X_t = y)$ であるので,

$$\begin{aligned}
\mathbb{P}^{L_y}_x(\cdot \cap L_y > 0) &= \int_0^\infty \mathbb{P}_x(L_y \in dt,\ L_y > 0)\mathbb{P}^{L_y}_x(\cdot \mid L_y = t) \\
&= \int_0^\infty dt \frac{p(t, x, y)}{S(b) - S(y)}\mathbb{P}^t_x(\cdot \mid X_t = y) \\
&= \frac{1}{S(b) - S(y)} \int_{x\vee y}^b S(dz)\mathbb{P}^{T_z}_x \circ (\mathbb{P}^{T_z}_y)^\wedge
\end{aligned}$$

最後の等号は Lemma 3 と Last Exit Time までにおける最大値は b 以上をとりえない事実から成立する. よって示せた. □

付録 4.B　MATLAB の実行コード

この節では当論文で実行した MATLAB のスクリプト等を掲載する．

4.B.1　逆ラプラス変換

```
"Scriptforlapinv.m"
% LapLOStd は 3 章で Explicit に表せた標準ブラウン運動におけるラプラス変換
% quadrature method を用いる．Cohen [2007] や Wang and Xu [2005] 等参照
clear
a=10^-6; % 逆変換における定数
T=10^-3; % 最終的にどのスケールで見るか
M=100; % 評価する点の数
N=100; % 評価する点の数
k=700;% Last Exit Time についてグラフをどの程度まで表示するかを表す指標
l=700; % Ocuupation Time についてグラフをどの程度まで表示するかを表す指標
h=2*T/N;
j=sqrt(-1);
I=40;
s1=zeros(10*N,1);
s2=zeros(10*M,1);
F=zeros(10*M,10*N);
% 必要な定数等の設定を行う

for n=1:10*N
for m=1:10*M
s1(n)=a+j*pi*(n-1)/N;
s2(m)=a+j*pi*(m-1)/M;
F(m,n)=LapLOStd(0,3,10,s1(n),s2(m));
% ラプラス変換からモーメント条件を作る
end
end

f=real(ifft2(F));
% モーメント条件の行列 F に対し 2 次元離散逆フーリエ変換を行う関数 ifft2 で評価
f1=zeros(k,l);
for m=1:k
for n=1:m
% m と n は等分されているので，m>n のところのみに値を代入していく
```

```
f1(m,n)=max(0,exp(a*m*T+a*n*T)*f(m,n)/M/h/pi);
% 元の時間軸のスケールに戻す，その際にマイナスになる部分は 0 としておく
end
end

 mesh(f1) % 3 次元グラフ化
 ylabel('last exit time');
 xlabel('occupation time');
 zlabel('f(l,a)');
```

4.B.2 $T_b - L_y$ と L_y のシミュレーション

```
"DriftTbL.m"
% Tb-Ly をシミュレーションで導出する
function [L]=DriftTbL(x, y,b,mu, sigma,dt, n)
counter=0; % Tb までのステップ数
lastcounter=0; % Last Exit するまでのステップ数 (離散なので最後に y より下にいた時刻)
Last=zeros(n,1); % 各サンプルにおける Last Exit の時刻ベクトル
X=x;
Tb=zeros(n,1); % 各サンプルにおいて Kill されるまでの時刻ベクトル
for i=1:n
while X < b % b に達したらブレイク
    X=X+mu*dt+sigma*sqrt(dt)*randn; % dt ごとに分布に従って過程を作っていく
    counter=counter+1;
    if X < y % y より下にいる場合はまだ last exit していないので現在のステップ数で更新
        lastcounter=counter;
    end
end
Tb(i)=counter*dt; % 時間のパーティションの幅をかけて時刻へと変換する
Last(i)=lastcounter*dt; % 上に同じ
end
L=mean(Tb-Last); % 期待値を取る
```

```
"DriftLast.m"
% Last Exit Time をシミュレーションで導出する
function [L]=DriftLast(x, y,b,mu, sigma,dt, n)
counter=0; % kill されるまでのステップ数
```

```
lastcounter=0; % Last Exit するまでのステップ数
Last=zeros(n,1); % Last Exit Time のサンプルベクトル
X=x; % X は dt 分過程を進めた時の値，x は初期値
for i=1:n
while X < b
    X=X+mu*dt+sigma*sqrt(dt)*randn;
    counter=counter+1;
    if X < y
        lastcounter=counter;
    end
end
Last(i)=lastcounter*dt; % 時間のパーティションの幅をかけて時刻へ変換
end
L=mean(Last); % 期待値を取る
```

```
"ScriptforSim.m"
x=0;
b=10;
mu=1;
sigma=1;
dt=10^-3;
n=10^5;
F=@(y)DriftTbL(x, y,b,mu, sigma,dt, n);% y に関する関数として定義する
G=@(y)DriftLast(x, y,b,mu, sigma,dt, n);
fplot(F, [x+10^-10, b-10^-10])
figure
fplot(G, [x+10^-10, b-10^-10])
```

4.B.3　費用最小化

```
"LapLO.m"
function[lap]=LapLO(mu, sigma, x, y,b,alpha,beta)
% ドリフト mu ≠ 0 のブラウン運動を扱う
% Last Exit Time と Occupation Time のラプラス変換を数値計算で求める関数
func2=@(z)Scalediff(mu,sigma, z)...
    .*LapFO(mu, sigma, x,y,z,alpha,beta).*LapFO(mu, sigma, y,y,z,alpha,beta);
    % 被積分関数を z で定義
```

```
    % LapFO は\citet{zhang2015occupation}の式をドリフト付ブラウン運動について書き下したもの
lap=1./(Scale(mu,sigma, b)-Scale(mu,sigma, max(x,y)))...
    .*integral(func2,max(x,y),b);% Theorem の式


"CostFunction.m"
function[Cost]=CostFunction(mu,sigma,x,y,b,alpha,beta)
Cost=1+LapLO(mu, sigma, x, y,b,alpha,beta)-LapLO(mu, sigma, x, y,b,0,beta);
% 費用関数を 4 章の具体例に即して作成


"plotcost.m"
function[]=plotcost(mu,sigma,x,b,alpha,beta)
% y に関して費用関数のグラフを作る関数
z=@(y)CostFunction(mu,sigma,x,y,b,alpha,beta);
fplot(z,[x,b-10^-10])


"ScriptforCostmin.m"
clear;
sigma=3;
x=0;
b=50;
n=10;
alpha=0.05;
beta=0.05;
mu=0.5;
plotcost(mu,sigma,x,b,alpha,beta)% mu を 0.5 としたときのグラフ
hold on % グラフを同じ figure 内に描画する
mu=1.5;
plotcost(mu,sigma,x,b,alpha,beta)% mu を 0.5 としたときのグラフ
mu=2.5;
plotcost(mu,sigma,x,b,alpha,beta)% mu を 0.5 としたときのグラフ
```

References

Bertoin, Jean (1996). *Lévy processes*. Vol. 121. Cambridge university press Cambridge.

Borodin, Andrei N and Paavo Salminen (2015). *Handbook of Brownian motion-facts and formulae*. Springer Science & Business Media.

Cohen, Alan M (2007). *Numerical methods for Laplace transform inversion*. Vol. 5. Springer Science & Business Media.

Csáki, Endre, Antónia Földes, and Paavo Salminen (1987). "On the joint distribution of the maximum and its location for a linear diffusion". In: *Annales de l'IHP Probabilités et statistiques*. Vol. 23. 2, pp. 179–194.

Dassios, Angelos (2005). "On the quantiles of Brownian motion and their hitting times". In: *Bernoulli* 11.1, pp. 29–36.

Egami, Masahiko and Rusudan Kevkhishvili (2017). "AN APPLICATION OF TIME REVERSAL TO CREDIT RISK MANAGEMENT". In: *arXiv preprint arXiv:1701.04565*.

Getoor, RK and MJ Sharpe (1973). "Last exit times and additive functionals". In: *The Annals of Probability* 1.4, pp. 550–569.

Karatzas, Ioannis et al. (1991). *Brownian motion and stochastic calculus*. Vol. 113. Springer Science & Business Media.

Pitman, Jim and Marc Yor (1982). "A decomposition of Bessel bridges". In: *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete* 59.4, pp. 425–457.

– (1996). "Decomposition at the maximum for excursions and bridges of one-dimensional diffusions". In: *Itô's stochastic calculus and probability theory*. Springer, pp. 293–310.

Rogers, Leonard CG and David Williams (2000). *Diffusions, markov processes, and martingales: Volume 1, foundations*. Vol. 1. Cambridge university press.

Salminen, P (1984). "One-dimensional diffusions and their exit spaces". In: *Mathematica scandinavica* 54.2, pp. 209–220.

Wang, Zewen and Dinghua Xu (Jan. 2005). "Numerical inversion of multidimensional Laplace transforms using moment methods". In: *Numerical Mathematics* 14.

Zhang, Hongzhong (2015). "Occupation times, drawdowns, and drawups for one-dimensional regular diffusions". In: *Advances in Applied Probability* 47.1, pp. 210–230.