

はじめに

金融ビジネスの世界で機械学習、ビッグデータ、AI、Fintechといったワードが飛び交うようになってずいぶん経ちます。学术界も例外ではありません。2019年11月にスタンフォード大学で開催されたシンポジウムでBrunnermeier et al. (2021) は機械学習のテクニックが従来は不可能だった金融市場の謎を解き明かす鍵になるのではないかと期待を込めたコメントをしています。2020年に発行された *The Review of Financial Studies* では機械学習に関する3本の論文が“New Methods in the Cross-Section”と題された特集号（第33巻第5号）に収録されました。また、Chen et al. (2019c) はFintechが金融産業に与えた影響を分析した論文を発表し、おおむね肯定的な評価を与えています。こうした中、ファイナンスにおける機械学習の応用やFintechに関する書籍もかなりの数が出版されるようになってきています。本書もその中の一つです。

これまでこの方面で出版されてきた書籍は筆者の知る限り大きく2つに分類できます。1つ目は、理論や技術に関する詳細には触れず機械学習やFintechの広範な活用を紹介するものです。2つ目は、読者がファイナンス理論も機械学習も基礎的な部分は知っていることを前提に書かれているものです。つまり、標準的なファイナンス理論については触れずに機械学習のテクニックを金融データへ応用した技術書です。

これらはいずれも有用ではありますし良書も多いです。しかし、初歩的な紹介に留まる本から学び始め、次に技術書に書かれた応用手法に取り組もうとする場合、大きなギャップがあるのも事実です。また、標準的なファイナンス理論に関する知識なしで技術書を読んでも機械学習のありがたみが今一つわからない可能性もあります。そこで本書はこうしたギャップを埋めることを目的の一つにしました。

まず標準的なファイナンス理論の骨格となる部分の説明をなるべく直観的な形で行いました。また極力、ファイナンス理論の自然な応用として機械学習の応用例を示すことを試みました。つまり、普通のファイナンス理論に関する教科書として読んでもらい、読み終えたらなんとなく機械学習の有用性が呑み込めている、という構成を目指しました。

そのため、本書で扱う機械学習の技術はいくぶん限定的です。機械学習は教師あり学習、教師なし学習、そして強化学習の三つに分類できると言われます。これらすべてが現在ファイナンスでは盛んに応用されていますが、本書では教師あり学習だけ（とはいえ、ディープラーニングまで）扱っています。また、教師あり学習を使ったファイナンスへの応用だけでもいまや多様な広がりを見せ

ているのですが、本書ではほぼ株価の予測に絞って議論をしています。ファイナンスの基礎理論との自然な接続を重視したからです。

もちろん、機械学習を扱う以上、プログラミングを避けて通るわけにはいきません。現在、機械学習の分野でプログラムを書く場合、Python がもっとも有力な選択肢の一つです。そこで、本書でも Python を使ったプログラミングを行います。

ただし、ファイナンスの理論的な部分を説明する際にはほぼ Python は使わないことにしました。これは教科書として本書を使う場合の汎用性を高めるためです。つまり、本書は次のような使い方を想定しました。

本書はファイナンス関連の講義テキストとして使用できます。その場合、第1章、第2章、第3章そして第5章、第6章のみ使うことでも一通りの内容をおさえることができます。大教室での講義に使いやすいよう、これらの章にはプログラムを含めていません。実際、筆者も勤務校で（本書のもととなった講義ノートを）このように使っています。

独力もしくは大学の演習科目の教材として本書に取り組む場合は最初から通読していただければと思います。ただし、第1章は本当に基本的な金融市場に関することを紹介しているだけなので、よく知っている読者は飛ばしても構いません。第Ⅰ部と第Ⅱ部は主に標準的なファイナンス理論を扱っています（第4章と第7章に関しては、どちらかといえば第Ⅲ部の準備といった意味合いが強いです）。第Ⅲ部は機械学習を使った株価の予測技術について議論しています。必要に応じてかいつまんで読むこともできますが、一応第Ⅱ部は第Ⅰ部を前提に、第Ⅲ部は第Ⅰ部と第Ⅱ部を前提に書かれています。

また、なぜ第Ⅲ部を置いたのかについては少し補足が必要だと思います。資産市場は時代が進み、技術と制度が整えられるにつれ効率的になっているとも言われます。しかし、実際に市場が効率的かどうかについては、今なお激しく議論が交わされるファイナンス理論における重要なトピックです。この問題は株価などの資産価格が予測できるかどうか、という議論に置き換えることもできます。もし本当に市場が効率的ならば株式などの資産価格の予測は不可能だからです。もちろん、仮に市場が非効率だとしても株価の予測は容易な作業ではありません。しかし近年では機械学習を用いた、いわゆるデータ駆動アプローチによる予測技術が発展しつつあります。そこで第Ⅲ部はデータ駆動アプローチと題し、機械学習を使った株価の予測手法について議論をします¹⁾。

ところで本書のほとんどの章では章末問題を置きましたが、これらには難問も含まれます。本文では説明しきれなかった理論を補足する意味合いの問題も多いです。難易度に応じて*や**といった印を付けたので、取り組んでみて難しいと感じたならばあまりこだわらずに解答を参照してください（同様に、本文でも少し発展的な内容を含む節や項には*印を付けました）。

本書はファイナンスに興味のある学部生、大学院生や社会人など幅広い読者を想定しています。ただ、数学があまりに苦手な読者にとっては苦痛かもしれません。この点について補足しておく、

¹⁾ とはいえ、比較的短時間で計算が終わるよう、サンプルサイズは抑えています。本書で示した結果については、機械学習の使い方に慣れるためのデモンストレーションと考えてください。

初等確率論や統計学のほか、行列や微分（テイラー展開やラグランジュ未定乗数法を含みます）の基本的な知識は必要です。経済学部や学部の生で中級以上のミクロ経済学を学んでいれば問題ないでしょう。理工系の学生や出身者にとってももちろん問題ないでしょう。しかし、もしこれらの知識に不安がある場合は、あらかじめ予習をしていただく必要があります²⁾。

Python についても環境は導入済みで基本的な操作方法は習得済みの読者を想定しています。Python の使い方について解説を行う章を一つ設けるべきかどうかは最後まで迷いました。しかし、どんなに丁寧に書いても公式のガイドを超えることができそうになかったので最終的には断念しました。もし Python の基本的な使い方について未修得の場合は公式ガイド (<https://www.python.jp/index.html>, 2022 年 7 月現在) に目を通しておいください。

次に Python の動作環境についてです。ハードウェアとしては Windows 10 Pro と macOS 11.1 と Ubuntu 20.04.2 で動作確認を行っています。これらに加えて、Google Collaboratory でも確認を行いました。Windows と Mac のスペックはいずれも CPU が Intel Core i7@ 2.30GHz, メモリーは 16GB です。Ubuntu のスペックは CPU が Intel Core i7@2.50GHz, メモリーは 7GB です³⁾。また Python および使用したパッケージのバージョンは下表の通りです。

	Windows	Mac	Ubuntu	GoogleColab
Python	3.7.9	3.8.5	3.7.11	3.7.12
pandas	1.2.1	1.1.3	1.3.4	1.1.5
pandas-datareader	0.10.0	0.10.0	0.10.0	0.9.0
matplotlib	3.3.2	3.3.2	3.5.1	3.2.2
scikit-learn	0.24.1	0.23.2	1.0.2	1.0.1
numpy	1.19.2	1.19.2	1.19.2	1.19.5
statsmodels	0.12.1	0.12.0	0.12.1	0.10.2
TensorFlow	2.3.0	2.3.0	2.4.1	2.7.0
python-dateutil	2.8.2	2.8.1	2.8.2	2.8.1

なぜこうしたことを細かくお伝えするかというと、特にディープラーニング用のパッケージである TensorFlow は少しのバージョン違いで動作しなくなることがあるからです。Python が新しすぎるだけで動かなくなることはもちろん、使用する関連パッケージのバージョン違いで動かなくなることすらあります。もちろん（筆者自身の転記ミスも含めて）プログラムの書き写し間違いや本文に記載した注意書きの見落としがあるかもしれません。しかし、プログラムが動かなかった場合はバージョン違いを疑ってみることを忘れないでください。

本書で紹介したプログラムとプログラムで用いる csv ファイルは次の GitHub のリンクから取得

²⁾ 岩波書店のキーポイント・シリーズなどは取り組みやすいと思います。ただし、『微分積分』（川村, 1996）、『線形代数』（薩摩・四ツ谷, 1992）、『確率・統計』（和達・十河, 1993）、『多変数の微分積分』（小形, 1996）など必要に応じて数冊読む必要があるのも、1冊で済ませたい場合は岩城 (2012) や加藤 (2018) などに取り組むのもよいと思います。

³⁾ 本書では GPU を使った学習は行っていない。

できます。

<https://github.com/dsk-yshkw/DataDrivenFinance>

最後になりますが、本書の執筆にあたって東京理科大学の岩城秀樹教授、UBS の Mikio Oba 氏にはプログラムも含めて本書のすみずみまで目を通していただき、有益なコメントをいただきました。筆者が北海学園大学経営学部と関西大学政策創造学部で行った演習授業の参加学生諸氏にもコメントをもらいました。また、共立出版の菅沼正裕氏には、本文に何度も目を通していただき有益な編集上の助言と訂正すべき誤りを指摘していただきました。本書を執筆する上で必要な資料や計算環境の整備にあたっては野村マネジメントスクールと科研費（課題番号 18K01694）に資金面でサポートしていただきました。この場を借りて御礼申し上げます。もちろん、本文中の誤りはすべて筆者の責任に帰すことはいうまでもありません。ご意見、ご指摘をいただければとてもありがたいです。

本書が読者の皆さんの好奇心を少しでも満たしてくれることを希望しています。

2022 年春 大阪にて

吉川 大介

目 次

第 1 章 金融の仕組みについて	1
第 I 部 均衡価格アプローチ	5
第 2 章 最適ポートフォリオ	6
2.1 期待リターンとボラティリティ	7
2.1.1 ポートフォリオの期待リターン	10
2.1.2 ポートフォリオのボラティリティ	12
2.1.3 期待リターンとボラティリティの関係	13
2.2 安全資産を含まないポートフォリオの最適化	15
2.3 安全資産を含むポートフォリオの最適化	17
第 3 章 CAPM	24
3.1 投資家の期待効用最大化	24
3.2 市場の均衡	29
3.2.1 超過供給と超過需要	29
3.2.2 CAPM	34
3.2.3* 確率的ディスカウント・ファクター	36
第 4 章 回帰モデル	39
4.1 正規方程式	39
4.2 モデルの推定	44
4.2.1 データの取得	44
4.2.2 前処理	49
4.2.3 データの分割	53
4.2.4 パラメータ推定	54
4.3 過学習	60
4.3.1 検証	60
4.3.2 正則化	64
4.3.3 グリッドサーチ	66

第 II 部 無裁定価格アプローチ	69
第 5 章 無裁定	70
5.1 定 義	71
5.2 無裁定とリスク中立確率	73
5.2.1 二項モデル	76
5.2.2 三項モデル	78
5.2.3 2 期間二項モデル	79
第 6 章 デリバティブ	84
6.1 先 渡	86
6.2 先 物	88
6.3 オプション	90
6.4 スワップ	94
第 7 章 統計的裁定	101
7.1 定 義	102
7.2* 共和分法	104
7.2.1 定常性と統計的裁定	104
7.2.2 共和分過程	106
7.2.3 Python による共和分検定	108
7.3 マーケット中立化法	119
7.3.1 裁定価格理論	120
7.3.2 裁定が可能な場合のマーケット中立化法	121
7.3.3* 統計的裁定が可能な場合のマーケット中立化法	122
7.4 デイスタンス法	123
第 III 部 データ駆動アプローチ	129
第 8 章 分 類	130
8.1 2 クラス分類	132
8.1.1 交差エントロピー誤差	132
8.1.2* 交差エントロピー誤差の含意	133
8.1.3 ロジスティック・シグモイド関数	134
8.1.4 ロジスティック回帰を用いた予測	136
8.2 多クラス分類	150
8.2.1 データの整備	152
8.2.2 フィッティング	154

第 9 章 ディープラーニング	157
9.1* ディープラーニングの基本的な仕組み	160
9.1.1 確率的勾配降下法	163
9.1.2 誤差逆伝播法	166
9.2 データの準備と Keras の基本	168
9.3 学習の高速化	175
9.3.1 早期終了	175
9.3.2 最適化手法	177
9.3.3 ミニバッチ	181
9.4 勾配消失問題	182
9.4.1 活性化関数	183
9.4.2 重みの初期値	186
9.4.3 バッチ正規化	187
9.5 過学習の抑制	189
9.5.1 正則化	189
9.5.2 ドロップアウト	190
 第 10 章 リカレントニューラルネットワーク	 193
10.1 データの準備	194
10.1.1 データの取得	194
10.1.2 共和分検定	196
10.1.3 系列データの整備	199
10.1.4 ARIMA	200
10.2 RNN	202
10.2.1 シンプルな RNN	202
10.2.2 LSTM	206
10.2.3 GRU	210
10.3 ディープラーニングの共和分法への応用	215
 参考文献	 221
章末問題解答	229
索引	258