

保険・年金 フォーカス

オープン・ソース型モデルの浸透 どのようにモデル統治を進めるべきか？

保険研究部 主席研究員 篠原 拓也
(03)3512-1823 tshino@nli-research.co.jp

1—はじめに

保険会社は、商品の価格設定や、準備金の積立・評価、必要資本の算定などのために、システム上にモデルを構築して、シミュレートすることが一般的だ。モデルを通じて、将来収支を計算したり、リスクの発生や影響度を見積もったりすることで、経営の意思決定やリスク管理に役立てている。

モデルの活用は、世界中の保険会社で行なわれている。従来は、保険会社内で最初から最後まですべてを作り込む、自社完結型のクローズド・ソース型のモデル構築が一般的であった。近年、さまざまな業界で見られるオープン・イノベーションの一環として、モデル構築においても、社外からの知見を取り入れるべく、オープン化を模索する動きが出ている。

本稿では、2018年にアメリカのアクチュアリー会冊子に掲載された記事をベースに、オープン・ソース型モデルや、その統治について検討していくこととしたい¹。

2—オープン・ソース型モデル

まず、オープン・ソース型モデルとはどういうものか、クローズド・ソース型モデルと比較しながら見ていくこととしよう。

1 | オープンソースは、誰でも自由に扱ってよいコンピュータープログラム

人間が理解しやすいプログラミング言語で書かれたコンピュータープログラムは、「ソースコード」と呼ばれる。コンピューターは、機械語で書かれたプログラムをもとに動いている。機械語のプログラムは、人間にとって理解が困難で、まるで暗号のようであることから、暗号を意味する「コード」という言葉を使ってプログラムのことを言い表すようになったといわれている。そして、人間がプログラミング言語で書いたコンピュータープログラムは、コードのもと(ソース)となるものであるため、ソースコードと呼ばれるようになった²。

「オープン・ソース」とは、そのソースコードを広く一般に公開して、誰でも自由に扱ってよいと

¹ “An Open Source” Rohan N. Alahakone, Dorothy L. Andrews (The Actuary, Jun. Jul. 2018, SOA)を参考にしている。

² ソースコードを機械語のプログラムに翻訳することをコンパイルと呼ぶ。

する考え方を指す。また、そうした考え方により公開されるソフトウェアのことを指す場合もある。

オープン・ソースは、アメリカのオープン・ソース・イニシアティブ³が、ソースコード形態での配布、ソースコードの改変・再頒布の自由、特定の人・集団・用途の差別禁止などの定義を置いている⁴。

2 | オープン・ソース型とクローズド・ソース型には、一長一短がある

多くの保険会社が、商品の価格設定をしたり、将来収支を計算したり、必要資本を算定したりする際のモデル構築において、オープン・ソース型とするか、クローズド・ソース型とするか、検討を始めているといわれる。モデルは、経営の意思決定や収支予測にとどまらず、財務や決算報告などでも、重要な役割を担う。もしモデルやその処理にエラーがあれば、多額の損失発生、法的制裁(罰金)、風評の失墜、はては財務破綻にまでつながる恐れがある。

オープン・ソース型とクローズド・ソース型には、一長一短がある。たとえば、ソースコードの管理という点では、クローズド型のほうが優れている。一方、ソースコードの透明性を確保する点では、オープン型のほうが上をいく。こうした長所や短所は、つぎのようにまとめることができる。

図表 1. クローズド型とオープン型の長所・短所

クローズド・ソース型		オープン・ソース型	
長所	短所	長所	短所
監査の記録が組み込まれる	新規導入や改変費用が高い	変更への柔軟性がある	監査の記録が残らない
版の管理が組み込まれる	カスタマイズや認証に追加費用がかかる	変更に必要な時間が短い	社内検証が必要
計算効率性の検証済み	ソースコードへのアクセスが制限される	提供者がカスタマイズを行わないため、費用があまり高額にならない	計算が非効率なことがある
ソースコードの変更が提供者により維持・管理される	ソース変更やカスタマイズが提供者に依存する	創造性やイノベーションを疎外しない	プログラミングのための人材訓練が必要
プログラムの正確性が提供者により検証済み	透明性に限界があるため、エラーの検出が遅れがち	透明性に限界がない	プログラムの変更が未承認となる恐れ 画一化のために標準プログラム技法が必要

※ “An Open Source” Rohan N. Alahakone, Dorothy L. Andrews (The Actuary, Jun. Jul. 2018, SOA) より、筆者作成。

3 | モデルリスクを減らすことがモデル構築の大きな課題

モデルを構築して活用する上で、考慮しなくてはならないリスクとして、「モデルリスク」が挙げられる。モデルリスクの定義にはいくつかのものがあるが、現在、比較的多くの人々が有効と考えている定義は、1996年にゴールドマン・サックスの定量戦略調査ノートで示されたものだ。そこでは、モデルリスクは「財務の意思決定を行なうためにモデルを利用することで、損失が生じるリスク」と定義されている。そしてこのノートでは、つぎのように、モデルリスクの種類が示されている。

³ オープンソース思想の啓蒙で有名なエリック・レイモンド氏らを中心に1998年に設立された非営利団体。

⁴ オープンソースの定義は、10カ条からなる。その項目は、再頒布の自由、ソースコード、派生ソフトウェア、作者のソースコードの完全性、個人または集団に対する差別の禁止、用途分野に対する差別の禁止、ライセンスの分配、特定製品でのみ有効なライセンスの禁止、他のソフトウェアを制限するライセンスの禁止、ライセンスの技術中立性。

図表 2. モデルリスクの種類

① モデルが適用できない
② モデルが誤っている
③ モデルは正当だが、解が誤っている
④ モデルは正当だが、利用法が不適切
⑤ 近似を誤っているモデル
⑥ ソフトウェアやハードウェアにバグがある
⑦ データが不安定

※ “Model Risk” Derman, Emanuel (Goldman Sachs Quantitative Strategies Research Notes, April 1996) より筆者作成。

モデルリスクを減少させるためには、「モデル統治」が必要となる。モデル統治とは、モデルの要素へのアクセス権限を特定の人に限定することや、要素変更の際に、社内の所定の手続きをとって、経営の了承を得ることなどを指す。

3—モデル統治のあり方

通常、オープン・ソース型モデルの場合、ソースコードが広く公開されていて、プログラム開発の柔軟性が高い。一方で、クローズド・ソース型モデルよりもモデルリスクが高まるという難点もある。そこで、「どのようにモデルを統治するか」がカギとなる。以下、モデル統治について、みていこう。

1 | まず、環境の設定、対象範囲の定義、担当者の職務設定が必要

モデル統治には、「モデル統治フレームワーク (Model Governance Framework, MGF)」を設けることが重要とされている。MGF は、環境の設定、対象範囲の定義、担当者の職務設定からなる。

(1) 環境の設定

統治を効果的に行なうには、全社の一貫した関与と、リスク管理のカルチャーの浸透が必要となる。一般に、新たなインフラを導入する時には、変化への抵抗を示す人々が出てくる。このため、MGF の目的や重要性について、すべての関係者を啓蒙することが重要となる。その際、関係者を開発に巻き込むことも有益となる。自らが開発したり、実行したりしたものには、支援する意識が働くためだ。このようにして、リスク管理のカルチャーが各部門や会社全体に自然に織り込まれていく必要がある。

(2) 対象範囲の定義

モデル統治の対象範囲には、会社機能に影響を及ぼすすべてのモデルが含まれる。しかし、モデル統治の初期段階では、MGF の対象となる重要なモデルをいくつか選ぶことが賢明となる。MGF では、一定の手順で処理が行なわれる。この手順には、モデルの妥当性確認、経営の承認などが含まれる⁵。

(3) 管理者の職務設定

モデル管理者の基本的な職務は、組織的に、MGF やその管理が遵守されていることの確認である。当初は、モデル管理者は、モデル開発者や利用者から良い評価が得られないことが多い。たとえば、「規約があるせいで、プログラム実行中の変更ができない」「モデル管理者がいることで、経営者が開発者や利用者を信用していないかのような印象を与える」

⁵ MGF 対象の手順として、仮定の設定、データの移動、モデルの高度化、モデルの妥当性確認、モデルの修正、モデルの保存記録、モデルの結果の利用、経営の承認、ソフトのアップデートと統合、ピア・レビューが挙げられる。

「モデルのプログラム変更よりも、打ち合わせや、細かい記述・分析作業ばかりが増加してしまう」
「文書化に多くの時間が必要となる」

といった声だ。しかし時を経て、モデル管理者の支援により統治手順が徐々に改良されてくると、開発者や利用者は、この枠組みがモデルリスクの軽減に効果があることに気づくようになる。

2 | モデルの関係者がそれぞれの役割を果たすことが重要

MGF を活用するには、モデルの関係者がそれぞれの役割を果たすことが重要となる。関係者には、モデル承認者、モデル開発者、モデル利用者、監視者、モデル管理者、保証人がいる。順次みていく。

(1) モデル承認者

モデル承認者は、プログラムや仮定の変更を承認する任務を負う。通常は、その管理を行うのは、上級経営管理者である。モデルの変更はすべて、承認される前に正当性を確認され、ピア・レビューを受け、文書化され、分析される。専門性や変更内容によっては、複数のモデル承認者グループが必要となる場合もある。たとえば、モデルの仮定を承認するグループとは別に、プログラムの変更を承認するグループが必要となることがある。

(2) モデル開発者

モデル開発者は、プログラムと仮定を変更する権限を付与されている。その専門性は、ソースプログラムと開発の知識に基づいている。プログラムの変更はすべて、一定のプログラミング基準にしたがって行われる。そして、モデル結果への影響が、計量され、確認され、分析され、独立して検証され、ピア・レビューされ、文書化される。モデル開発者は、保証人(後述)の要求にしたがってモデル変更に取り組み、承認を受けるために、モデル承認者に変更内容を提示する。

(3) モデル利用者

モデル利用者は、報告や分析の目的でモデル結果を利用する。具体的には、準備金評価、価格設定、GAAP 財務報告、エンベディッド・バリュー報告、将来予測などに責任を負っている。モデル利用者は、モデル開発者と緊密に連携して、モデルの変更に関するさまざまな要求を提出する。また、開発の前には、モデル機能のピア・レビューに深く関与することが多い。

(4) 監視者(ゲートキーパー)

監視者は、開発モデルの管理人である。監視者は、開発領域に、承認されたモデル変更だけが導入されることを確認するために、モデル管理者と緊密に連携する。その特徴的な役割は、モデルを最新のバージョンに置き換えるにあたって、それまで使用していた古い開発のバージョンを保管することにある。そして、新しいバージョンが所期したとおりに稼動することを確認する。監視者は、開発環境に正当なモデルが導入されることを確保する任務を負っている。

(5) モデル管理者

モデル管理者は、モデル統治の中心的な役割を演じる。すべての人がモデル統治の枠組みに沿って機能することを確認する。モデル管理者の責任には、モデル変更の過程がスムーズに進行することの確認が含まれる。これは、承認者との打ち合わせを進めたり、すべてのモデル変更要請のリストを作成してスケジュールを立案・管理したり、モデル変更要請の優先順位付けをしたり、あらゆる集団と一緒に要請に対する最適解が得られることを確保したりすることなどによって行われる。モデル管理

者は、適切に評価を行ない、オープン・ソース型のソフトウェアをよく理解しておくことが期待されている。プロジェクト管理のスキルは、その役割に不可欠のものとなる。

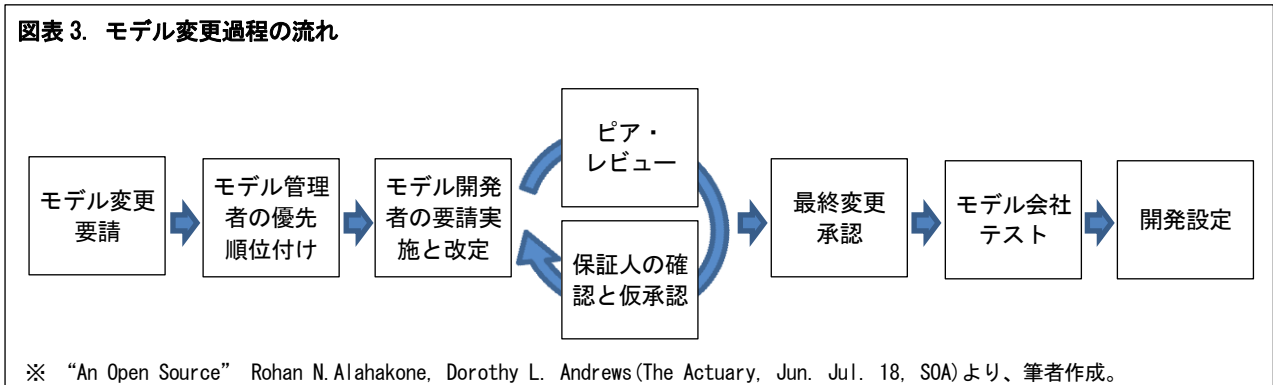
(6) 保証人(スポンサー)

モデル変更の保証人は、通常、諸準備金評価、GAAP 等の財務報告領域、もしくは価格設定について権限と責任をもつ上級管理職が務める。保証人は、モデル管理者とともにモデルの変更要請のスケジュールを立てる。保証人は、モデル変更要請が予定通りに実施されることを確認する職務を負う。

3 | モデルの変更においては、ルールの管理、任務の切り分けが重要

(1) モデル変更過程

モデル変更は、つぎの図の流れにしたがう必要がある。



上記変更過程の中で、途中の段階をとばすことは許されない。それぞれの過程における厳格さの程度は、各領域ごとに最低基準を決めて設定される。

- プログラミング：一定のプログラミング基準にしたがって行われる
- ピア・レビュー：変更内容に基づいて、適度な厳格さが求められる
- 分析：変更内容の複雑さに応じて、適度な厳格さが求められる
- 確認：測定方法に基づいて、認容性の水準が定められる
- 文書化：目的に応じて厳格さの水準が定められる

(2) 任務の分離

関係者は、1つの役割だけをこなすべきである。これは、他の役割を担う際に生じる基準の希薄化や役割の間の衝突を避けるためだ。各関係者の役割と責任は、慎重かつ明確に定められる必要がある。それらは、正式に文書化されて、いつでも遵守されることが必要となる。

(3) 情報技術(IT)の活用

ITは、モデル統治成功のカギとなる。つぎの事項の確認のためにIT部門との連携が必要となる。

- すべての人が、ファイルフォルダ(ディレクトリ)とネットワークに正しいアクセス権や制限を有している
- ソフトウェアのライセンスの更新とアップデートが、スムーズに時間通りに行われる
- モデルの枠組みやソフトウェアに対して、専従のIT専門職がいる
- 災害時の復旧とバックアップの過程が、あらかじめ設定されている
- 監査に対して、常にITの過程管理が機能している

4—モデル統治の効果

モデル統治がうまくいくと、つぎのようなメリットがもたらされる。

—モデル開発者は、管理された環境下でモデル変更の作業が進められることの満足感を持つ。自分の役割を効率よく果たすために、MGFを維持することを学び、その結果、興奮や満足感を得ることができる。

—モデル管理者は、継続的に保険計理や法定実務の進歩とともに管理や過程を改善する。役割が明確に決められていることで、より効果的に、モデル統治の枠組みに関与しながら職務を遂行することができる。

—組織は、モデルリスクが大きく軽減される。それは、モデルに対する上級管理職の信頼感を高め、併せて、モデル関係者間のモデル保持に対するプライドを高める。また、MGFによるリスクの軽減は、オープン・ソース型とクローズド・ソース型のシステムの間の多くの反対意見を取り除く。

組織は、両者それぞれの性能を互角に評価したり、費用対効果を比較したりすることができる。

なお、MGFの企画や実行は、組織ごとに異なる。MGFは、既存の枠組み、カルチャー、組織の特性に応じて、企画されるべきである。もし上級管理職がモデル統治を切望して、これにうまく関与すれば、組織に費用対効果の改善というメリットがもたらされるだろう。

5—おわりに（私見）

従来、モデルの構築・運用は、企画部門やリスク管理部門、システム部門など、会社内の限られた組織内で行われてきた。そこでは、モデルリスクの管理は関係者の間だけで対処することができた。

しかし、オープン・ソース型モデルが導入されれば、その有用性が高まるとともに、関係者が増えて、モデルリスクも増大する。そこで、モデル統治やそのための枠組み(MGF)を設けることが重要となる。

保険会社のモデル統治の考え方や、MGFの構築は、まだ緒についたばかりといえる。今後、保険業界でモデルの構築がどのように変化していくのか、またMGFがどのように組み立てられていくのか、興味深い。引き続き、その動向に注目していくこととしたい。