# アクチュアリー業務における新たな分析・解釈手法の活用
# ～位相的データ解析（TDA）および最大解釈分解（MID）～
# ＜ASTIN 関連研究会・データサイエンス関連基礎調査部会＞

Shift Technology Japan　小田　秀匡
EY ストラテジー・アンド・コンサルティング　汪　沁亮
SOMPO リスクマネジメント　松森　至宏
T&D ホールディングス　浅芝　良一

**【司会】**　時間となりましたので、セッション B-5、ASTIN 関連研究会およびデータサイエンス関連基礎調査部会による、「アクチュアリー業務における新たな分析・解釈手法の活用～位相的データ解析および最大解釈分解～」を開始します。

発表者は、前半が Shift Technology Japan の小田さんと EY ストラテジー・アンド・コンサルティングの汪さん、後半が SOMPO リスクマネジメントの松森さんと T&D ホールディングスの浅芝さんの合計 4 名です。なお、質疑応答の時間は、前半、後半、それぞれの後に取らせていただきます。また、Slido に投稿された質問に対しても、その際に回答することにいたします。オンラインで視聴されていて質問のある方は、前半、後半の各発表時間中に Slido への投稿をお願いします。それでは、小田さん、汪さん、よろしくお願いします。


**【小田】**　はい、小田です。よろしくお願いいたします。前半は、位相的データ解析、Topological Data Analysis のお話をします。私のパートでは、基本的な定義と性質を説明させていただいて、汪の方から応用をお話しさせていただきます。

## 概要

位相的データ解析（Topological Data Analysis; TDA）は点群（有限次元空間内の有限個の点; point cloud）を分析対象とする数理手法である。主に３次元空間データ（例：タンパク質構造）の形状把握や時系列データ（例：金融・感染症）の傾向把握・異常検知に応用がある。

本発表では、位相的データ解析の数理的に初歩的な内容を解説し、実データ（特に保険データ）への応用の可能性を探る。



位相的データ解析は数理手法の 1 つなのですけれども、分析対象としているものは点群といって、有限次元空間内の有限個の点になります。主に 3 次元空間データや時系列データ、そういったものに応用があります。

## 保険分野での応用の可能性 ①

- 不正検知
  - 背景: 保険金請求や顧客行動データには、典型的なパターンと異常なパターンが混在している。従来の統計モデルでは検出が難しい場合がある。
  - TDA の役割: 請求データを高次元の点群（顧客属性、請求額、タイミングなど）として捉え、そこから得られる位相的特徴（クラスター構造や循環構造）を調べることで、異常なパターン（不正の兆候）を検出できるかもしれない。

4 / 44

私が知る限り、保険業界で具体的に TDA が使われているという話は知らないのですけれども、考えられる応用というものをいくつか紹介させていただきます。最初は、不正検知で応用が考えられるかと思っています。保険金請求や顧客行動のデータで特異なパターンのようなものが存在している場合に、TDA の位相的な特徴を使うことによって、不正な兆候を検出できるかもしれない。こういった応用が 1 つあるかと思います。

## 保険分野での応用の可能性 ②

- 死亡率・罹患率のクラスタリング
  - 背景: 年齢別・地域別・疾病別の死亡率や罹患率データには、見えにくい共通パターンが存在する。
  - TDA の役割: 多次元データを解析することで、似た動きをするサブグループを自動的に発見し、従来のクラスター分析より柔軟に「リスク構造」を可視化できるかもしれない。

5 / 44

次に、死亡率や罹患率のクラスタリングにも応用があるのかと思っていて、いくつか共通パターンがあると思うのですけれども、そういったもの見つけてくるなど、少し変わった行動をしているようなデータ点を見つけられる可能性があるのではないかというところでも、応用が 1 つあるかと思います。

## 保険分野での応用の可能性 ③

- 自然災害リスク・カタストロフィーモデル
  - 背景: 台風・洪水・地震などの災害リスクは空間的・時間的に複雑な依存関係を持っている。
  - TDA の役割: 気象データや地理空間データに対して「異常パターン」や「空間構造」（例: 洪水の広がり方、台風の経路パターン）を抽出できるかもしれない。これにより、災害リスクの評価やモデリングの改善が期待できるかもしれない。

6 / 44

さらに、自然災害周りでも応用があるかと思っていて、台風にせよ、洪水にせよ、地震にせよ、災害リスクというものは空間的な広がりと時間的な広がりの両方を持っています。そういったものに関して、いくつかのフェーズや異常パターンのようなものを見つけることができるかもしれない。これは保険分野と関係なく、先行研究がありまして、保険分野の興味としては、例えば保険料率の算定などの見積もりに応用があるかもしれないと考えられているということですね。

## 保険分野での応用の可能性 ④

- ポートフォリオの多様化分析
  - 背景: 生命保険会社や損保会社は、資産運用リスクを分散する必要がある。しかし、相関構造は時間とともに変化し、従来の分散分析だけでは把握しづらい。
  - TDA の役割: 金利曲線や株価の時系列の市場状態の「形」を捉え、通常時とストレス時の違いを可視化できるかもしれない。

7 / 44

それから、資産運用サイドでも応用が少しあるかと思っていまして、各種金融資産、そういったものの値動きは、多次元の時系列データですけれども、そのようなものの構造ですね。時間変化や暴落に代表されるようなショックを見つけたり、あるいはそういった兆候を事前に見つけたりすることができるかもしれない。この観点では、私の発表の後、汪の方から具体的な話があるかと思います。

## 保険分野での応用の可能性 ⑤

- 契約者行動（解約・継続）のパターン分析
  - 背景: 保険契約者の解約、更新、商品変更といった行動は複雑な時系列データである。
  - TDA の役割: 顧客ごとの行動履歴を解析することで、似た行動を取るグループや異常行動を識別でき、マーケティングやリスク管理に活用できるかもしれない。

8 / 44

それから、マーケティングの方にも少し応用があるかと思っていて、保険契約や更新や解約ですね。そういったところで応用があるかもしれないと、今、考えています。クラスタリングや、他には、やはり特異な行動をしている顧客を見つけるなど、そういったところに役割があるのではないかと思っています。

本日の発表は、 最新の研究手法を紹介することは目的とせず、
2010 年代に提唱された Persistence Landscape 等の基本的な事項について紹介する。

文献 [Bubenik, 2015]
*Bubenik, P. (2015). Statistical topological data analysis using persistence landscapes. J. Mach. Learn. Res., 16(1), 77-102.*

(数学の専門用語の日本語訳は [Hiraoka, 2013] を参考にした。)

発表の最後に、実際の金融データに基づく簡単なクラスタリングの結果を紹介する。

9 / 44

本日は、特に最新の手法・研究を紹介する、具体的な保険分野への応用を話すということは目的としていなくて、非常に基本的な内容です。今から 10 年以上前に提唱された、Persistence Landscapes などの基本的な事項について説明します。

10 / 44

まず、少し数理的な話をしないといけないと思っていて、Persistence Landscape というものはPersistence Module に対して定義されるのですけれども、Persistence Module というものは、これは単純に数学の話なのですけれども、説明しておきます。

## Persistence Module

実数体 $\mathbb{R}$ 上の persistence module とは全順序集合 $(\mathbb{R}, \leq)$ から実線型空間の圏への共変関手である。

**定義 (persistence module)**

実数体 $\mathbb{R}$ 上の線型空間 $M_r$ の族 $\{M_r\}_{r\in\mathbb{R}}$ と線型写像 $M(r_1 \leq r_2)\colon M_{r_1} \to M_{r_2}$ の族 $\{M(r_1 \leq r_2)\}_{r_1\leq r_2}$ の組が (実数体 $\mathbb{R}$ 上の) persistence module であるとは、

- 任意の $r \in \mathbb{R}$ に対して $M(r \leq r)$ が恒等写像であり
- 任意の $r_1 \leq r_2 \leq r_3$ に対して $M(r_1 \leq r_2) \circ M(r_2 \leq r_3) = M(r_1 \leq r_3)$ が成立する

ことである。

ただし、式中 $(r_1, r_2) \in \{(r_1, r_2) \in \mathbb{R}^2 \mid r_1 \leq r_2\}$ を $r_1 \leq r_2$ と略記している。以下同様。

11 / 44

一般に、半順序集合から加群の圏への関手のことを Persistence Module というのですけれども、今日の発表ではそれほど抽象的な話をするつもりはないので、要するに、半順序集合としては実数体ですね。全順序集合である実数体を考えてもらって、加群の圏としては実線型空間の圏を考えていただければ結構です。つまり、何か実数 R に対応して、$M_r$ という線型空間が対応していて、もし、$r_1$、$r_2$ という実数が 2 つあれば、値の小さい方から値が大きい方に向かって射が存在している。さらに、次のようないくつかの公理を満たしているときに、Persistence Module といいます。これは単純に、純粋に、今、算数の話です。

Persistence module $M$ から persistence module $N$ への射は線型写像 $f_r : M_r \to N_r$ の族 $\{f_r\}_{r\in\mathbb{R}}$ であって、任意の $r_1 \leq r_2$ に対して $f_{r_2} \circ M(r_1 \leq r_2) = N(r_1 \leq r_2) \circ f_{r_1}$ を満たすものである。

Persistence module の圏は Abel 圏[1] である [Bubenik and Scott, 2014]。

Persistence module の具体的な構成には様々なものが知られている。本日の発表では、典型的かつ実用性があると考えられる、Čech 複体から構成される persistent homology と Vietoris-Rips 複体から構成される persistent homology を紹介する。

[1] Homology 代数の議論を展開するのに十分な構造を備えた加法圏

12 / 44

Persistence Module 間の射というものは、$M_r$ から $N_r$ の線型写像の族です。次式を満たすもので、Persistence Module から Persistence Module への射を定義してやる。そうすると、Persistence Module の圏が Abel 圏になるということが知られていて、したがって、その上で homology 代数が展開できる。ただ、これだけだと抽象的で全然実感がわかないと思いますので、いくつか具体的な構成というものを紹介します。今日は、Čech 複体から構成される persistent homology と、Vietoris-Rips 複体から構成される persistent homology を紹介します。



13 / 44

## 単体複体

### 定義 (単体複体)

$X$ を有限集合とする。$\Sigma$ を「$X$ の部分集合」の集合とする。$\Sigma$ が (抽象) 単体複体であるとは、

- 任意の $x \in X$ に対して $\{x\} \in \Sigma$ であり
- 任意の $\tau \in \Sigma$ と任意の $\tau$ の部分集合 $\sigma$ に対して $\sigma \in \Sigma$ であることである。

単体複体 $\Sigma$ の元で特に $k+1$ 個の $X$ の元から構成されるものを $k$-単体という。

### 例

$X = \{1,2,3\}$ に対して $\Sigma = \{\{1\},\{2\},\{3\},\{1,2\},\{2,3\},\{1,3\}\}$ は単体複体である。$\{1\}$ は 0-単体であり $\{1,2\}$ は 1-単体である。

14 / 44

単体複体も、少し説明しておく必要があると思うのです。単体複体というものは、与えられた集合の部分集合の集合のことで、いくつかの公理を満たすものをいうのですけれども、すべての頂点が入っていて、もし、その単体が単体複体の中に入っていれば、その面も全部入っている。そのような条件を満たすときに、単体複体という。単体複体に含まれている元、つまり、X の部分集合なのですけれども、k＋1 個の元から構成されているときに、k-単体といいます。例えば、X が 1、2、3 という 3 つの数字からなる集合であれば、$\Sigma$ というものは単体複体の定義を満たしている。なぜかというと、すべての頂点が入っていて、$\{1,2\}$は 1 と 2 を結ぶ辺だと思っておくと、端点である 1 と 2 も両方入っていますから、ちゃんと単体複体になっている。$\{1\}$、$\{2\}$、$\{3\}$は 0 次元の単体だと思っているから、これは 0-単体と言い、$\{1,2\}$は 1 次元の単体だと思っているので、1-単体と言いますということです。

$X = \{x_1, \cdots, x_m\}$ を $\mathbb{R}^N$ 内の有限個の点の集合とする。各 $r > 0$ に対して単体複体 $\Sigma(X_r)$ を構成する方法を示す。単体複体の族 $\{\Sigma(X_r)\}_{r>0}$ が集合 $X$ の幾何的な情報を保持していることに期待する。

$r > 0$ を固定する。各点 $x_i \in X$ に対し半径 $r$ の閉球 $B_r(x_i) := \{x \in \mathbb{R}^N \mid \|x - x_i\| \leq r\}$ を考える。閉球の族 $\{B_r(x_i)\}_{i=1,\cdots,m}$ は集合 $X$ の被覆を与えている。 この集合 $X$ の被覆 $X_r := \coprod_{i=1}^m B_r(x_i)$ から単体複体 $\Sigma(X_r)$ を構成することを考える。

15 / 44

それはいい。では、それをわれわれが分析しようとしている点群の分析にどのように使うか、応用するかということなのです。もし、有限個の点があれば、その有限個の点を中心とした閉球のようなものを考えていただくと、その閉球というものは元の点の被覆を与えているわけですね。この被覆から単体複体を構成する方法がいくつか知られている。だから、有限個の点が与えられていて、半径 r が 1 つ固定されたら、その情報から 1 つ単体複体が作れるという話を今からします。

## Čech 複体

引き続き $r > 0$ を固定し、簡単のため、集合 $X = \{x_1, \cdots, x_m\}$ を $X = \{1, \cdots, m\}$ と表記し、閉球 $B_r(x_i)$ を $B_i$ と表記する。Čech 複体は集合 $X$ の被覆 $\{B_i\}_{i=1,\cdots,m}$ の脈体として定義される。

### 定義 (Čech 複体)

集合 $X = \{1, \cdots, m\}$ に対して、その単体複体 $\Sigma$ を

$$\{i_0, i_1, \cdots, i_k\} \in \Sigma \iff \bigcap_{j=0}^{k} B_{i_j} \neq \emptyset$$

で定める。単体複体 $\Sigma$ を Čech 複体と呼ぶ。

16 / 44

まず、Čech 複体というものを説明します。これが一番有名な、単体複体の構成かと思っていて、もし、過去に聞いたことがあるという人は、この構成方法を見たことがあるかと思うのです。閉球が、今、いくつかあるのですけれども、例えば、$i_0$、$i_1$ から $i_k$ というものは、元に与えられた集合の部分集合なわけだけれども、これが、まずその単体複体に入っているかということは、閉球がすべて交わっているところがあるか。もしあれば、それをその単体として認識しましょう。このようにして。単体複体を作る方法があります。このように作ったものを、Čech 複体といいます。これが一番自然な定義です。

そうなのですけれども、これは k 個の共通部分があるかどうかを調べないといけないから、実際、プログラムを書こうと思うと結構大変ですね。

## Vietoris-Rips 複体

3 つ以上の閉球の交わりの検証は計算量的に困難である。そこで Čech 複体の亜種 (簡易版) として Vietoris-Rips 複体が知られている。

### 定義 (Vietoris-Rips 複体)

集合 $X = \{1, \cdots, m\}$ に対して、その単体複体 $\Sigma$ を

$$\{i_0, i_1, \cdots, i_k\} \in \Sigma \iff B_{i_s} \cap B_{i_t} \neq \emptyset \text{ for } 0 \leq s, t \leq k$$

で定める。単体複体 $\Sigma$ を Vietoris-Rips 複体と呼ぶ。

17 / 44

そこで、データ分析の分野では、こちらの Vietoris-Rips 複体というものがよく使われています。この考え方は、Čech 複体とよく似ているのですけれども、計算量が圧倒的に下げられます。どのようにやるかというと、先ほどはこの k 個の点、k 個のボールの共通部分があるかどうかで認識していたのですけれども、任意の 2 個のボールに交わりがなければ、それで単体を認識しましょうという方法です。これだと、2 点間の距離さえ求まれば重なっているかどうかが計算できるので、計算量がだいぶ落ちます。そうですね、データ分析という観点では、大体この方法が使われることが多いかと思います。

今、Čech 複体と Vietoris-Rips の両方を紹介したのですけれども、違いがどちらもよく分からないという人もいると思うので、差が出る例を見ておきましょう。

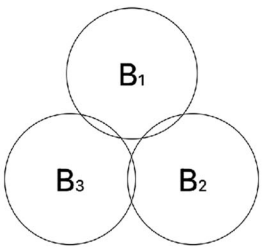



Figure: $\mathbb{R}^2$ 内の 3 点 $X = \{1, 2, 3\}$ の場合

### 例 (Čech 複体)

$\Sigma = \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}\}$

### 例 (Vietoris-Rips 複体)

$\Sigma = \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}, \{1, 2, 3\}\}$

18 / 44

今、3 つの閉球を描いてあるのですけれども、どの 2 つも重なっています。$B_1$ と $B_2$ で重なりがあるし、$B_2$ と

$B_3$に重なりがあるし、$B_3$と$B_1$にも重なりがあるので、Vietoris-Rips 複体の意味では 2-単体です。1、2、3 の頂点で囲まれたこの三角形の部分を認識する。けれども、Čech 複体の場合は、$B_1$、$B_2$、$B_3$すべてが共通で交わっているところがないですね。真ん中に穴が空いていますので、3 つの球すべてが交わっているわけではないので、ここには 2-単体を認識しません。そういったところから、差が出てきますというわけです。もし、ここに穴が空いているということを認識したのであれば、この Čech 複体を使うしかないわけです。だから、Vietoris-Rips 複体を考えるということは、ここに空いている穴を無視するということになります。

## 鎖群

一般に、単体の頂点の置換を考えることにより、単体の向き付けを考えることができる。向きの指定された単体 $\{1,\cdots,k\}$ を $\langle 1,\cdots,k\rangle$ のように表記することにする。例えば、1-単体の場合は $\langle 1,2\rangle = -\langle 2,1\rangle$ といった具合である。

単体複体に含まれる向きづけられた $k$-単体が生成する実数体 $\mathbb{R}$ 上の線型空間を $k$-鎖群 $C_k$ という[2]。負の整数 $k$ に対しては $C_k = 0$ とする。$k$-鎖群 $C_k$ の元を $k$-鎖という。

[2]実数体 $\mathbb{R}$ ではなく整数環 $\mathbb{Z}$ や剰余環 $\mathbb{Z}/2\mathbb{Z}$ 上で生成される自由加群を考える場合もある。

19 / 44

なぜ、このような話をしたのだろうか。単体複体で定義できると、どうして Persistence Module が定義できるのかというところを説明していく必要があり、そのために、まず、鎖群という概念を説明する必要がある。単体があると、単体の向きを付けることができますね。例えば、k 個の点があれば、遇置換であれば同じ向き、奇置換であれば違う向きというように考えてやると、単体に対して向き付けをすることができる。だから、単体複体に含まれる、向き付けられた k-単体が生成する、線型空間のようなものを考えてやることができる。そのように作ったものを k-鎖群といいます。ここでは線型空間なのですけれども、英語だと chain group ですね。そのように向き付けられた k-単体が張る線型空間のことを、鎖群といいます。

## 鎖複体

向き付けられた $k$-単体 $\langle 0, 1, \cdots, k\rangle$ に対して

$$\partial_k\langle 0, 1, \cdots, k\rangle = \sum_{i=0}^{k}(-1)^i\langle 0, 1, \cdots, \check{i}, \cdots, k\rangle$$

と定め、線型に拡張して線型写像 $\partial_k\colon C_k \to C_{k-1}$ を得る。線型写像 $\partial_k$ を $k$-鎖群 $C_k$ 上の境界作用素という。例えば、
$\partial_2\langle 1, 2, 3\rangle = \langle 1, 2\rangle + \langle 2, 3\rangle + \langle 3, 1\rangle$ である。

各 $k$ に対して $\partial_{k-1} \circ \partial_k = 0$ が成立する。

鎖群と境界作用素からなる系列 $(C_k, \partial_k)$ を単体複体の鎖複体という。

20 / 44

そうすると、その鎖群に対して境界作用素というものを定義してやることができる。例えば k-単体があれば、このような式で k-1 単体に射が作れます。こういったものが生成しているということなので、線型に伸ばしてやれば、k-鎖群から k-1 鎖群に写像が得られる。この線型写像のことを、境界作用素や境界準同型という。式で見ると分かりにくいのですけれども、言っていることは非常に単純で、境界を取りますということです。2-単体があれば、$\langle 1, 2\rangle$ という 1-単体と、$\langle 2, 3\rangle$ という単体と、$\langle 3, 1\rangle$ になって、境界を取っていますね。

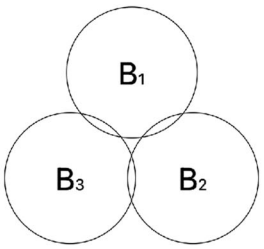



Figure: $\mathbb{R}^2$ 内の 3 点 $X = \{1, 2, 3\}$ の場合

**例 (Čech 複体)**

$\Sigma = \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}\}$

**例 (Vietoris-Rips 複体)**

$\Sigma = \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}, \{1, 2, 3\}\}$

18 / 44

先ほどの図だと、1、2、3 という 2 単体があったときに境界はこの周りですから、1 から 2 に行って、2 から 3 に行って、3 から 1 に行くわけですから、この境界作用素というものは、まさに境界を取っているわけです。

## 鎖複体

向き付けられた $k$-単体 $\langle 0,1,\cdots,k\rangle$ に対して

$$\partial_k\langle 0,1,\cdots,k\rangle = \sum_{i=0}^{k}(-1)^i\langle 0,1,\cdots,\check{i},\cdots,k\rangle$$

と定め、線型に拡張して線型写像 $\partial_k\colon C_k \to C_{k-1}$ を得る。線型写像 $\partial_k$ を $k$-鎖群 $C_k$ 上の境界作用素という。例えば、
$\partial_2\langle 1,2,3\rangle = \langle 1,2\rangle + \langle 2,3\rangle + \langle 3,1\rangle$ である。

各 $k$ に対して $\partial_{k-1}\circ\partial_k = 0$ が成立する。

鎖群と境界作用素からなる系列 $(C_k,\partial_k)$ を単体複体の鎖複体という。

20 / 44

境界を取って、もう 1 回境界を取るとなくなりますね、縁の縁はないですから。だから、2 回やると消えるというわけです。k-鎖群の系列というものは、複体になっている。一般に 2 回やると消えるというような系列のことを複体といいますけれども、このようにしてやると、鎖複体というものを作ることができます。英語でいうと chain complex です。はい。

## Homology 群

鎖複体 $(C_k,\partial_k)$ が与えられているとする。

各 $k$ に対し、$k$-鎖群 $C_k$ の部分加群（部分線型空間）$Z_k, B_k$ を次で定める。

$$Z_k := \mathrm{Ker}\partial_k\,,$$
$$B_k := \mathrm{Im}\partial_{k+1}\,.$$

各 $k$ に対し、$\partial_{k-1}\circ\partial_k = 0$ なので $B_k \subset Z_k$ である。

商加群（商線型空間）$H_k := Z_k/B_k$ を $k$-次 homology 群という。また、$k$-次 homology 群の元を $k$-次 homology 類という。

21 / 44

一般に複体があると、homology 群を定義することができて、どうやって定義するかというと、kernel を image で割ってやると、これもまた商加群、この場合は、線型空間の構造を持つのですけれども、こうやって割った空間のことを homology 群といいます。k-次 homology 群の元のことを、k-次 homology 類と言いますというわけです。

## Persistent Homology

$X = \{x_1, \cdots, x_m\}$ を $\mathbb{R}^N$ 内の有限個の点の集合とする。

各 $r > 0$ に対して、集合 $X$ の被覆 $\{B_r(x_i)\}_{i=1,\cdots,m}$ を考え、その単体複体 $\Sigma(X_r)$ を構成する方法を示し、その homology 群 の定義を与えた。$r > 0$ に対応する $k$-次鎖群と $k$-次 homology 群をそれぞれ $C_k(X_r)$, $H_k(X_r)$ と表記することにする（これらを、被覆 $X_r := \coprod_{i=1}^m B_r(x_i)$ または図形 $X_r := \bigcup_{i=1}^m B_r(x_i)$ の要約量だと考えると分かりやすい)。$r < 0$ に対しては $C_k(X_r) = C_k(X_0)$, $H_k(X_r) = H_k(X_0)$ とする。

各 $k$ に対して、実数体 $\mathbb{R}$ 上の線型空間の族 $\{H_k(X_r)\}_{r\in\mathbb{R}}$ が persistence module の構造を持つことを示す。



では、なぜそれが最初に説明した Persistent Homology と関係してくるのかということなのです。まず、今、実数値 r を 1 つ固定したら、それに従って homology 群が定義できるという話をしたのですけれども、今度は r を動かしてみましょうということを考えるわけです。だから、今、各 r に対して、その homology 群の定義ができるけれども、それを r に添えづけられた family だと思いましょう。そうすると、これが実は Persistence Module の構造を持つということを説明しております。

$r_1 \leq r_2$ とする。

単体複体 $\Sigma(X_{r_1})$ は単体複体 $\Sigma(X_{r_2})$ に埋め込まれている（部分集合である）ので、各 $k$ に対して、$k$-鎖群の埋込（単射線型写像）$\iota_k \colon C_k(X_{r_1}) \to C_k(X_{r_2})$ が誘導される。

更に、各 $k$ に対して $\iota_k \circ \partial_k = \partial_k \circ \iota_k$ が成立するので、 $k$-鎖群の埋込 $\iota_k \colon C_k(X_{r_1}) \to C_k(X_{r_2})$ は $k$-次 homology 群間の準同型（線型写像）$\iota_{*k} \colon H_k(X_{r_1}) \to H_k(X_{r_2})$ を誘導する。$\iota_k$ が単射であっても誘導された $\iota_{*k}$ は単射とは限らないことに注意する。

$M_r := H_k(X_r)$、$M(r_1 \leq r_2) := \iota_{*k}$ とすれば、$k$-次 homology 群が persistence module になることが分かる（Persitence module の定義 (p. 11) 参照)。これを単体複体から構成された persistent homology と呼ぶ。



もし、$r_1$ が $r_2$ より小さければ、$r_1$ から生成された単体複体は、$r_2$ から生成された単体複体に埋め込まれています。それはいいと思います。半径が大きくなれば半径が大きくなるほど、より重なるわけですから、半径が大きくなっていくと、どんどん単体複体が含まれていくわけですから、これは単調に大きくなっていくわけです。だから、元々埋め込みがあるわけです。境界準同型とは可換になりますね。縁を取ってから埋め

込むというのと、埋め込んでから縁を取るということは同じですね。だから、境界準同型と可換になるということは、homology 群間の準同型を導くということなので、導かれた射というものが定義できるわけです。$r_1$ と $r_2$ にこのような大小関係があれば、homology 群の方にも同じ向きで射が誘導される。当然、元の射が単射であったからといって、誘導される射が単射になるとは限らないのですけれども、とにかく射が誘導されるわけです。

## Persistence Module

実数体 $\mathbb{R}$ 上の persistence module とは全順序集合 $(\mathbb{R}, \leq)$ から実線型空間の圏への共変関手である。

### 定義 (persistence module)

実数体 $\mathbb{R}$ 上の線型空間 $M_r$ の族 $\{M_r\}_{r\in\mathbb{R}}$ と線型写像 $M(r_1 \leq r_2)\colon M_{r_1} \to M_{r_2}$ の族 $\{M(r_1 \leq r_2)\}_{r_1\leq r_2}$ の組が (実数体 $\mathbb{R}$ 上の) persistence module であるとは、

- 任意の $r \in \mathbb{R}$ に対して $M(r \leq r)$ が恒等写像であり
- 任意の $r_1 \leq r_2 \leq r_3$ に対して $M(r_1 \leq r_2) \circ M(r_2 \leq r_3) = M(r_1 \leq r_3)$ が成立する

ことである。

ただし、式中 $(r_1, r_2) \in \{(r_1, r_2) \in \mathbb{R}^2 \mid r_1 \leq r_2\}$ を $r_1 \leq r_2$ と略記している。以下同様。

11 / 44

もう 1 回、Persistence Module の定義に立ち返ると、その Persistence Module というものは、線型空間の族と線型写像の族の組み合わせだったわけだけれども、このように作ったものが、こういった公理を満たすということはチェックしてもらうと分かると思うので、

$r_1 \leq r_2$ とする。

単体複体 $\Sigma(X_{r_1})$ は単体複体 $\Sigma(X_{r_2})$ に埋め込まれている（部分集合である）ので、各 $k$ に対して、$k$-鎖群の埋込（単射線型写像）$\iota_k\colon C_k(X_{r_1}) \to C_k(X_{r_2})$ が誘導される。

更に、各 $k$ に対して $\iota_k \circ \partial_k = \partial_k \circ \iota_k$ が成立するので、 $k$-鎖群の埋込 $\iota_k\colon C_k(X_{r_1}) \to C_k(X_{r_2})$ は $k$-次 homology 群間の準同型（線型写像）$\iota_{*k}\colon H_k(X_{r_1}) \to H_k(X_{r_2})$ を誘導する。$\iota_k$ が単射であっても誘導された $\iota_{*k}$ は単射とは限らないことに注意する。

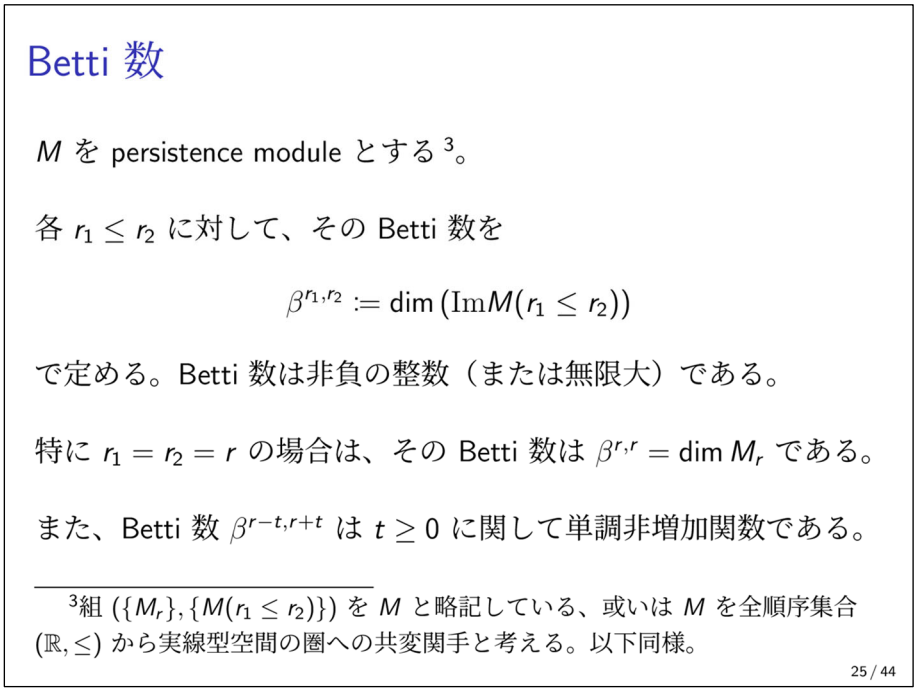
$M_r := H_k(X_r)$、$M(r_1 \leq r_2) := \iota_{*k}$ とすれば、$k$-次 homology 群が persistence module になることが分かる（Persitence module の定義 (p. 11) 参照）。これを単体複体から構成された persistent homology と呼ぶ。

23 / 44

実は、この Persistent homology というものは、Persistence Module の構造を持つということが知られてい

る。ぱっと分からなかった人は、少し手で計算をしてみてください。このように、単体複体から構成された Persistent homology と呼びます。


Persistence Landscape の話を今日はしたいのです。なぜ、Persistence Module があると、Persistence Landscape という概念が定義できて、それが大事になるのかという話をしておきます。




また、少し複雑な話になってしまって申し訳ないのですけれども、もし Persistence Module があれば、その Persistence Module に対して Persistence Landscape が定義できるという話をしておく。もし、$r_1$、$r_2$ で $r_1$ の方が小さいという関係があれば、Betti 数というものがこのように定義できる。元々、$r_1$ が $r_2$ より小さければ、$r_1$ から $r_2$ に従って、その射の image の dimension を計算することができて、それを Betti 数と呼んでいる。これは次元だから、非負の整数ですね。無限大かもしれないけれども、基本的に非負の整数です。

## Persistence Module

実数体 $\mathbb{R}$ 上の persistence module とは全順序集合 $(\mathbb{R}, \leq)$ から実線型空間の圏への共変関手である。

### 定義 (persistence module)

実数体 $\mathbb{R}$ 上の線型空間 $M_r$ の族 $\{M_r\}_{r\in\mathbb{R}}$ と線型写像 $M(r_1 \leq r_2)\colon M_{r_1} \to M_{r_2}$ の族 $\{M(r_1 \leq r_2)\}_{r_1\leq r_2}$ の組が (実数体 $\mathbb{R}$ 上の) persistence module であるとは、

- 任意の $r \in \mathbb{R}$ に対して $M(r \leq r)$ が恒等写像であり
- 任意の $r_1 \leq r_2 \leq r_3$ に対して $M(r_1 \leq r_2) \circ M(r_2 \leq r_3) = M(r_1 \leq r_3)$ が成立する

ことである。

ただし、式中 $(r_1, r_2) \in \{(r_1, r_2) \in \mathbb{R}^2 \mid r_1 \leq r_2\}$ を $r_1 \leq r_2$ と略記している。以下同様。

11 / 44

もし $r_1$ と $r_2$ が一緒であれば、もう 1 回、Persistence Module の定義に立ち返ると、恒等写像ですということも入っているので、これは $M_r$ の次元に一致するわけです。

## Betti 数

$M$ を persistence module とする[3]。

各 $r_1 \leq r_2$ に対して、その Betti 数を

$$\beta^{r_1,r_2} := \dim\left(\mathrm{Im} M(r_1 \leq r_2)\right)$$

で定める。Betti 数は非負の整数（または無限大）である。

特に $r_1 = r_2 = r$ の場合は、その Betti 数は $\beta^{r,r} = \dim M_r$ である。

また、Betti 数 $\beta^{r-t,r+t}$ は $t \geq 0$ に関して単調非増加関数である。

[3]組 $(\{M_r\}, \{M(r_1 \leq r_2)\})$ を $M$ と略記している、或いは $M$ を全順序集合 $(\mathbb{R}, \leq)$ から実線型空間の圏への共変関手と考える。以下同様。

25 / 44

また、$r_1$ と $r_2$ の差が小さいとき、$r_1$ として r-t を取って、$r_2$ として r+t を取ったときに、この t に対して単調非増加になるということも、

## Persistence Module

実数体 $\mathbb{R}$ 上の persistence module とは全順序集合 $(\mathbb{R}, \leq)$ から実線型空間の圏への共変関手である。

### 定義 (persistence module)

実数体 $\mathbb{R}$ 上の線型空間 $M_r$ の族 $\{M_r\}_{r\in\mathbb{R}}$ と線型写像 $M(r_1 \leq r_2)\colon M_{r_1} \to M_{r_2}$ の族 $\{M(r_1 \leq r_2)\}_{r_1\leq r_2}$ の組が (実数体 $\mathbb{R}$ 上の) persistence module であるとは、

- 任意の $r \in \mathbb{R}$ に対して $M(r \leq r)$ が恒等写像であり
- 任意の $r_1 \leq r_2 \leq r_3$ に対して $M(r_1 \leq r_2) \circ M(r_2 \leq r_3) = M(r_1 \leq r_3)$ が成立する

ことである。

ただし、式中 $(r_1, r_2) \in \{(r_1, r_2) \in \mathbb{R}^2 \mid r_1 \leq r_2\}$ を $r_1 \leq r_2$ と略記している。以下同様。

11 / 44

元々の Persistence Module のこれに入っていた、こちらの条件ですね。2 番目の方の条件を読み解けば、t に関して単調非増加だということもすぐに分かるかと思います。

## Persistence Landscape

$[-\infty, \infty]$ を $\overline{\mathbb{R}}$ と表記することにする。Persistence module $M$ の persistence landscape $\lambda$ とは実数直線 $\mathbb{R}$ 上で定義された $\overline{\mathbb{R}}$-値関数 $\lambda_k$ の列 $\lambda = (\lambda_1, \lambda_2, \cdots)$ である。或いは、$\{1, 2, \cdots\} \times \mathbb{R}$ 上で定義された $\overline{\mathbb{R}}$-値関数 $\lambda$ と考えても良い。

### 定義 (Persistence landscape)

各 $k = 1, 2, \cdots$ に対して、実数直線 $\mathbb{R}$ 上の $\overline{\mathbb{R}}$-値関数 $\lambda_k$ を

$$\lambda_k(r) := \sup\left\{t \geq 0 \mid \beta^{r-t,r+t} \geq k\right\} \quad \text{for } r \in \mathbb{R}$$

で定める。

どのような $r \in \mathbb{R}$ に対しても $\lambda_k(r) \geq 0$ ではあるが、後程 persistence landscapes の和差を考えたいので、関数 $\lambda_k$ は $\overline{\mathbb{R}}$-値関数と考えることにする。

26 / 44

そうすると、Persistence Landscape というものは、このように定義してやることができる。Persistence Landscape は、Persistence Module が与えられればそれは計算できるもので、実数直線上で定義された実数値関数なのです。だから、実数値関数の列なのです。k という添え字でやっていますけれども、あるいは $\lambda_1$、$\lambda_2$、$\lambda_3$ という列のことを Persistence Landscape と呼んでいて、その 1 つ 1 つの Persistence Landscape は、実数直線上の関数である。それをどうやって計算するかというと、この $\beta$ の r-t と r+t が落ちる瞬間があるのですけれども、その落ちる瞬間までを計っているというものが、Persistence Landscape なのです。これは式だけを見るとぱっと分かりにくいと思うので、少し図や具体的な例を見ながら説明していきますね。

Persistence module $M$ が単体複体から構成された persistence homology の場合（例えば 1-次 Homology 群 $H_1$）を考えると状況を理解しやすい。

この場合は、$t = 0$ で $\beta^{r,r} < +\infty$ は homology 群の次元に他ならず、$t \to +\infty$ では $\beta^{r-t,r+t} \to 0$ である。

したがって、homology 群の次元 $\beta^{r,r}$ が正の場合は、Betti 数が非負の整数であることに注意すれば、必ずどこかの $t > 0$ で Betti 数 $\beta^{r-t,r+t}$ の値が非連続に落ちる瞬間がある。Persistence landscape $\lambda$ は、その瞬間までの距離（時間）を計測している。

27 / 44

もし、Persistence Module が 1-次 homology であれば、もう少し分かりやすいかと思っていて、もし t がゼロであれば、その $\beta^{r,r}$ というものは homology 群の次元に他ならなくて、大きくすると 0 に落ちます。どこかで必ず落ちるので、そのどこか落ちるところが Persistence Landscape なのですけれども、これでも少し、まだ意味が分かりにくいですね。もう少し具体的な例で見ていきましょう。

## 1-次 Homology 群 $H_1$ の場合

$\mathbb{R}^2$ 内の点群に対して Čech 複体を計算した例 [Bubenik, 2015]。

3 個の 1-次 homology 類の生成と消滅が観測できる。

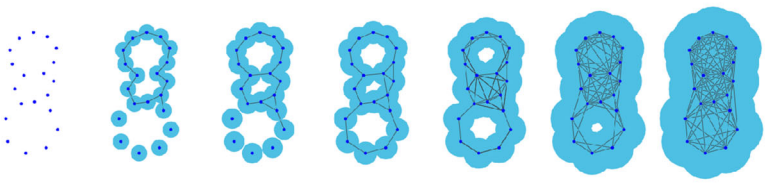


Figure 1: A growing union of balls and the 1-skeleton of the corresponding Čech complex. As the radius grows, features—such as connected components and holes—appear and disappear. Here, the complexes illustrate the births and deaths of three holes, homology classes in degree one. The corresponding birth-death pairs are plotted as part of the top left of Figure 2.

28 / 44

こちらは、Bubenik の論文から取ってきた図なのですけれども、これが一番分かりやすいかと思います。今、2 次元の空間の中に点がいくつかあって、この点が分析対象になっています。だんだんと半径を大きくしていっている様子が見えている。半径がもしゼロであれば、輪っかはどこにもないですけれども、少し大きくなるとくっついて、ここに輪っかが 1 個できています。もう少し半径が大きくなると、ここととここに 2 つ輪っかができていて、もう少し大きくなると、3 個になっている。もう少し半径が大きくなると、真ん中の輪

っかがつぶれてなくなって、もう少し半径が大きくなると、一上の輪っかがなくなって、もう少し半径が大きくなると、全部輪っかが消えましたという状態なのです。これだと、簡単ですけれども、数式に落とし込むことができないわけです。そのようなことを落とし込むために、Persistence Module の概念があるわけです。

## Persistence Diagram

各 homology 類に対して、生存が観測される $r$ の範囲 $(r_{\text{birth}}, r_{\text{death}})$ を算出し、それらを二次元平面に描写したものを persistence diagram と呼ぶ。

通常、位相的データ解析の分野では、この persistence diagram が点群の要約量として用いられ、分析の対象になることが多い。

次ページに、前ページで紹介した Čech 複体の persistence diagram（左上）と persistence landscape（左下）を示す [Bubenik, 2015]。

29 / 44

このようなときに、Persistence Landscape がどうなるかという説明をするのですけれども、その前に Persistence Diagram の話をしておきたい。Persistence Diagram はどのようなものかというと、各 homology 群の生存が観測される r の範囲を算出している。r がある範囲から、ある範囲までの間に生存しているというものがありますね。

## 1-次 Homology 群 $H_1$ の場合

$\mathbb{R}^2$ 内の点群に対して Čech 複体を計算した例 [Bubenik, 2015]。

3 個の 1-次 homology 類の生成と消滅が観測できる。

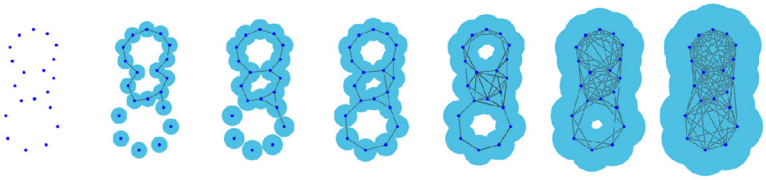


Figure 1: A growing union of balls and the 1-skeleton of the corresponding Čech complex. As the radius grows, features—such as connected components and holes—appear and disappear. Here, the complexes illustrate the births and deaths of three holes, homology classes in degree one. The corresponding birth-death pairs are plotted as part of the top left of Figure 2.

28 / 44

例えばこの輪っかであれば、ここのときに生成されて、ここのときに死滅しているので、ここからここま

でが生存している時間だということなのです。

## Persistence Diagram

各 homology 類に対して、生存が観測される $r$ の範囲 $(r_{\text{birth}}, r_{\text{death}})$ を算出し、それらを二次元平面に描写したものを persistence diagram と呼ぶ。

通常、位相的データ解析の分野では、この persistence diagram が点群の要約量として用いられ、分析の対象になることが多い。

次ページに、前ページで紹介した Čech 複体の persistence diagram（左上）と persistence landscape（左下）を示す [Bubenik, 2015]。

29 / 44

生まれたときと死んだときの r の、ここですね。2 次元にプロットしたもののことを Persistence Diagram と呼んでいる。通常、位相的データ解析の分野では、この Persistence Diagram は点群の要約量として用いられることが多くて、分析の対象になることが多いです。

## 1-次 Homology 群 $H_1$ の場合

$\mathbb{R}^2$ 内の点群に対して Čech 複体を計算した例 [Bubenik, 2015]。

3 個の 1-次 homology 類の生成と消滅が観測できる。

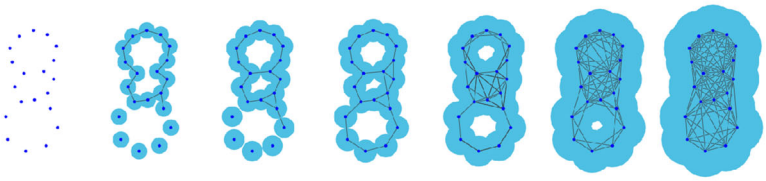


Figure 1: A growing union of balls and the 1-skeleton of the corresponding Čech complex. As the radius grows, features—such as connected components and holes—appear and disappear. Here, the complexes illustrate the births and deaths of three holes, homology classes in degree one. The corresponding birth-death pairs are plotted as part of the top left of Figure 2.

28 / 44

実際に、先ほどの例です。この例で、Persistence Diagram を描いてみると、このような感じになっている。

## Persistence Diagram と Persistence Landscape

[Bubenik, 2015]

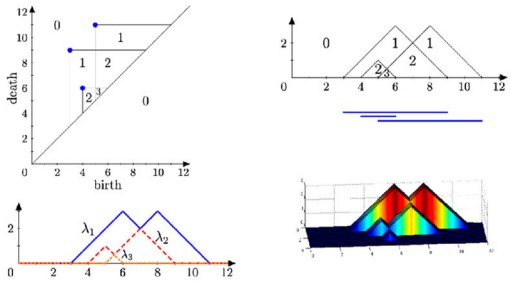



Figure 2: Persistence landscapes for the homology in degree 1 of the example in Figure 1. For the rank function (top left) and rescaled rank function (top right) the values of the functions on the corresponding region are given. The top left graph also contains the three points of the corresponding persistence diagram. Below the top right graph is the corresponding barcode. We also have the corresponding persistence landscape (bottom left) and its 3d-version (bottom right). Notice that $\lambda_1$ gives a measure of the dominant homological feature at each point of the filtration.

30 / 44

この左上の図を見てください。この 3 つの点が描かれていて、例えばここにしましょうか。ここで見ると、半径 3 のときですね。この x 軸が生まれたときの r、y 軸の方が死んだときの r なのですけれども、ここの点に着目すると、半径 3 のときにできて、半径 9 のときに死んでいると出ているので、

## 1-次 Homology 群 $H_1$ の場合

$\mathbb{R}^2$ 内の点群に対して Čech 複体を計算した例 [Bubenik, 2015]。

3 個の 1-次 homology 類の生成と消滅が観測できる。

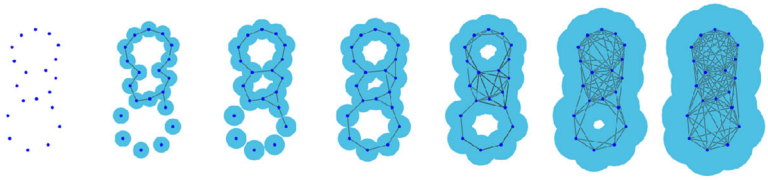


Figure 1: A growing union of balls and the 1-skeleton of the corresponding Čech complex. As the radius grows, features—such as connected components and holes—appear and disappear. Here, the complexes illustrate the births and deaths of three holes, homology classes in degree one. The corresponding birth-death pairs are plotted as part of the top left of Figure 2.

28 / 44

それがこの輪っかです。今、半径 3 のときに生まれて、ここですね。ここが半径 9 なのですけれども、半径 9 のときに死んだということで、

## Persistence Diagram と Persistence Landscape

[Bubenik, 2015]

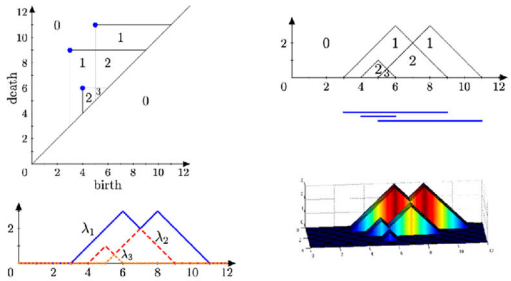



Figure 2: Persistence landscapes for the homology in degree 1 of the example in Figure 1. For the rank function (top left) and rescaled rank function (top right) the values of the functions on the corresponding region are given. The top left graph also contains the three points of the corresponding persistence diagram. Below the top right graph is the corresponding barcode. We also have the corresponding persistence landscape (bottom left) and its 3d-version (bottom right). Notice that $\lambda_1$ gives a measure of the dominant homological feature at each point of the filtration.

30 / 44

それをここの点に描いているわけです。

次に、ここを見ましょうか。4、6 のところにプロットがあると思うのですけれども、4 のときに生まれて、6 のときに死んだというものが出ていて、

## 1-次 Homology 群 $H_1$ の場合

$\mathbb{R}^2$ 内の点群に対して Čech 複体を計算した例 [Bubenik, 2015]。

3 個の 1-次 homology 類の生成と消滅が観測できる。

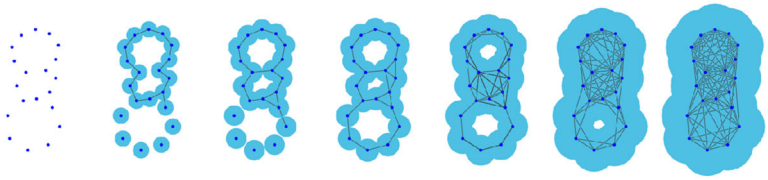


Figure 1: A growing union of balls and the 1-skeleton of the corresponding Čech complex. As the radius grows, features—such as connected components and holes—appear and disappear. Here, the complexes illustrate the births and deaths of three holes, homology classes in degree one. The corresponding birth-death pairs are plotted as part of the top left of Figure 2.

28 / 44

それがここです。半径 4 のときにでき上がって、半径 6 のときにつぶれてなくなっています。

## Persistence Diagram と Persistence Landscape

[Bubenik, 2015]

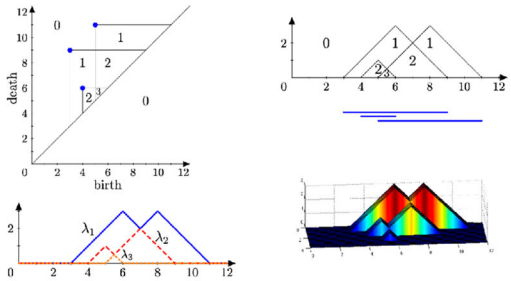



Figure 2: Persistence landscapes for the homology in degree 1 of the example in Figure 1. For the rank function (top left) and rescaled rank function (top right) the values of the functions on the corresponding region are given. The top left graph also contains the three points of the corresponding persistence diagram. Below the top right graph is the corresponding barcode. We also have the corresponding persistence landscape (bottom left) and its 3d-version (bottom right). Notice that $\lambda_1$ gives a measure of the dominant homological feature at each point of the filtration.

30 / 44

最後に、ここです。ここが半径 5 のときにでき上がって、半径 11 のときに消えている。

## 1-次 Homology 群 $H_1$ の場合

$\mathbb{R}^2$ 内の点群に対して Čech 複体を計算した例 [Bubenik, 2015]。

3 個の 1-次 homology 類の生成と消滅が観測できる。

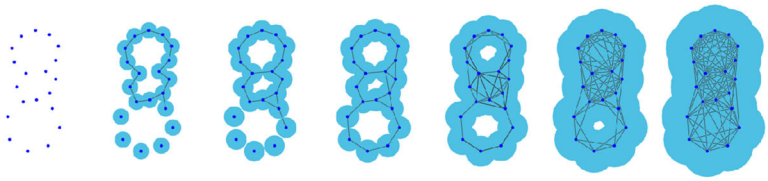


Figure 1: A growing union of balls and the 1-skeleton of the corresponding Čech complex. As the radius grows, features—such as connected components and holes—appear and disappear. Here, the complexes illustrate the births and deaths of three holes, homology classes in degree one. The corresponding birth-death pairs are plotted as part of the top left of Figure 2.

28 / 44

ここか。これか。半径 5 のときにできて、半径 11 のときに消えている。3 つの 1 次元 homology 類が見えていて、その 3 つの 1 次元 homology 類を今ここに描いているわけです。

## Persistence Diagram と Persistence Landscape

[Bubenik, 2015]

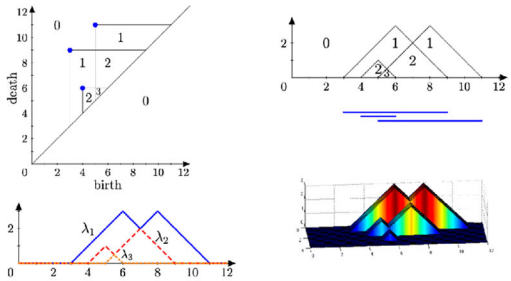



Figure 2: Persistence landscapes for the homology in degree 1 of the example in Figure 1. For the rank function (top left) and rescaled rank function (top right) the values of the functions on the corresponding region are given. The top left graph also contains the three points of the corresponding persistence diagram. Below the top right graph is the corresponding barcode. We also have the corresponding persistence landscape (bottom left) and its 3d-version (bottom right). Notice that $\lambda_1$ gives a measure of the dominant homological feature at each point of the filtration.

30 / 44

Persistence Diagram というものはそれだけなのですけれども、よく補助線として 45 度線を引いてあることがあると思っていて、45 度線をどうして引いてあるかというと、この 45 度線の上に乗っている homology 類というものは、生成したタイミングと死滅したタイミングが一致しているということで、生まれた瞬間に消えているということが分かります。もしかするとノイズなどで生まれたのかもしれないというわけで、顕著な homology 類とはみなされないわけです。逆に、45 度線から離れているということは、生まれてから死ぬまで結構時間がたっている、長く生存したということを意味している。それは、顕著なループの存在を意味しているわけで、45 度線から離れていれば離れているほど、大事な輪っかなのではないかと普通考えるわけです。これが、Persistence Diagram の説明なのです。

実はこの図を 45 度、時計回りに回転してもらうと、こちらの下の図が出てくると思うのです。実は、これが Persistence Landscapes になっています。この Persistence Diagram に縦線と横線の補助線が引いてあるのですけれども、それも込みで 45 度回転してもらうと、このような図形が出てくると思うのです。1 番上のこの青い線、これが実数値関数だと思ってください。実数直線上に定義されている関数だと思って、これが、まず $\lambda_1$ である。次に、この線です、この線が $\lambda_2$、ここが $\lambda_3$ です。こう行って、こう行って、こう行く。これが $\lambda_3$。$\lambda_4$ 以降は、まっすぐ恒等的に 0 の線になっている。だから、Persistence Diagram から Persistence Landscapes を作ることができて、Persistence Diagram というものはただの点の描写なのですけれども、Persistence Landscapes というものは関数列だ。実数直線上で定義された実数値関数列だということになるわけです。

実はこれは逆もできて、Persistence Landscapes を計算してあると、実はそこから Persistence Diagram を復活させることができる。ということは、持っている数理的情報と一緒なのではないか。わざわざ Persistence Landscapes を考える必要はないのではないかと思う人がいるかもしれないですけれども、実はそうではないのです。

前ページで見たように、persistence diagram と persistence landscape の数理的情報は等価である。しかしながら、その数理的な取り扱いの良さは異なる。

Persistence landscape は関数空間に値を取る。関数空間には自然に線型空間としての構造が備わり、適切な位相を設定することにより完備距離空間 (Banach 空間) になる。実際に観測された persistence landscape を、persistence landscape に値を取る確率変数の 1 つの実現（標本）と見る立場に立てば、persistence landscapes の標本平均や期待値を定義することができる。

一方、persistence diagram が値を取る空間には、Banach 空間のような数学的取扱が良い構造を付与することは困難である。このような背景から persistence diagram に代わり persistence landscape が提唱されるに至った [Bubenik, 2015]。

31 / 44

Persistence Diagram と Persistence Landscapes は持っている数理的情報は等価なのですけれども、しかし、数理的取り扱いのよさは結構違う。

## Persistence Diagram と Persistence Landscape

[Bubenik, 2015]

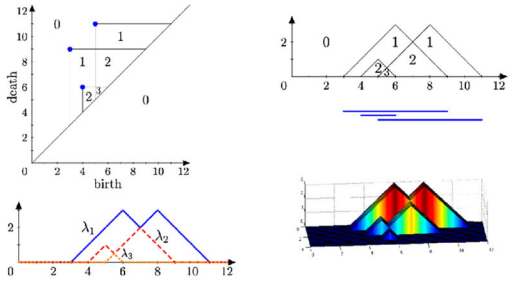



Figure 2: Persistence landscapes for the homology in degree 1 of the example in Figure 1. For the rank function (top left) and rescaled rank function (top right) the values of the functions on the corresponding region are given. The top left graph also contains the three points of the corresponding persistence diagram. Below the top right graph is the corresponding barcode. We also have the corresponding persistence landscape (bottom left) and its 3d-version (bottom right). Notice that $\lambda_1$ gives a measure of the dominant homological feature at each point of the filtration.

30 / 44

なぜかというと、Persistence Diagram というものは点を打っているだけなので、例えば 2 つあって、片方は 3 つの点があって、片方は 2 つの点があると、中間の状態のようなものが定義できないですね。3 つの点と 2 つの点の中間の状態はないので、中間の状態を定義できない。ということは、標本平均や期待値のようなものも計算できない。

前ページで見たように、persistence diagram と persistence landscape の数理的情報は等価である。しかしながら、その数理的な取り扱いの良さは異なる。

Persistence landscape は関数空間に値を取る。関数空間には自然に線型空間としての構造が備わり、適切な位相を設定することにより完備距離空間 (Banach 空間) になる。実際に観測された persistence landscape を、persistence landscape に値を取る確率変数の 1 つの実現（標本）と見る立場に立てば、persistence landscapes の標本平均や期待値を定義することができる。

一方、persistence diagram が値を取る空間には、Banach 空間のような数学的取扱が良い構造を付与することは困難である。このような背景から persistence diagram に代わり persistence landscape が提唱されるに至った [Bubenik, 2015]。



けれども、Persistence Landscape は関数空間に値を取るので、中間の状態や標本平均などが定義できるのです。関数空間でも、当然、自然に線型空間としての構造が備わっていて、さらに適切な位相を設定してやると完備距離空間になる。つまり、Banach 空間になる。ここは結構大事なところです。だから、実際に観測された Persistence Landscape を、Persistence Landscape に値を取る確率変数の 1 つの標本だと思う立場に立てば、Persistence Landscape の標本平均や期待値を定義することができるわけです。

一方、Persistence Diagram が値を取る空間は、かろうじて距離空間にはなるのですけれども、完備距離空間にはならないということが知られていて、特に Banach 空間にはなりません。こういった背景から、Persistence Diagram に代わって、Persistence Landscape が提唱されるに至ったわけです。

## Persistence landscapes の空間

Persistence landscapes の空間には自然に $L^p$-norm が定義され Banach 空間になる。

$$\|\lambda\|_p := \left( \sum_{k=1}^{\infty} \|\lambda_k\|_p^p \right)^{\frac{1}{p}} \quad \text{for } 1 \leq p < +\infty\,,$$
$$\|\lambda\|_\infty := \operatorname{ess\,sup} \lambda = \|\lambda_1\|_\infty$$

Banach 空間に値を持つ確率変数に対しては、その積分[4] が定義できる場合があり、独立同分布に従う確率変数列に対する「大数の強法則」や「中心極限定理」が成立する条件が知られている。

[4] 弱積分 (Pettis integral)



実は、Persistence Landscapes の空間は、$L^p$-norm を定義することによって Banach 空間になることが分か

ります。Banach 空間に値を取ると何がうれしいのかということなのですけれども、Banach 空間に値を取れば、実は積分が定義できる場合がある。そうすると、期待値が定義できる場合があるのです。逆に言うと、そうでないと、積分や期待値のようなものを数学的に定義することが極めて難しくなってしまう。さらに Banach 空間に値を取る場合は、独立同分布に従う確率変数列に対する「大数の強法則」や「中心極限定理」が成立する条件も知られているというわけで、非常に数学的扱いはよいわけです。

そもそも、Banach 空間に値を取る確率変数に対して、どうやって大数の強法則や中心極限定理を定義するのですかということなのですけれども、基本的には汎関数を経由して定義することになるのです。

特に実応用上重要な汎関数は

$$\lambda \mapsto \left( \sum_{k=1}^{K} \left\| \chi_{[-R,R]}\, \lambda_k \right\|_p^p \right)^{\frac{1}{p}}$$

である。ただし、$K$ は正の整数、$R$ は正の実数である。また、$\chi_{[-R,R]}$ は区間 $[-R, R]$ の指示関数である。
計算機で persistence landscape $\lambda$ の $L^p$-norm を計算するにあたっては、実数列 $\|\lambda_1\|_p^p, \|\lambda_2\|_p^p, \cdots$ 全てを足し合わせることはできず、事前に決め置いた $K$ 番目までの和を取って算出するより他なく、また各 $\lambda_k$ の $p$-norm も全範囲 $[-\infty, +\infty]$ で積分することはできず、事前に決め置いた $[-R, R]$ の範囲で積分して算出するより他ない。しかし、そのような簡便な方法で算出した $L^p$-norm に対しても、「大数の強法則」や「中心極限定理」が成立する条件が知られており、大変有意義なことである。

33 / 44

その中でも、実応用上特に重要な汎関数は次のようなもので、要するに $L^p$-norm なのですけれども、

## Persistence landscapes の空間

Persistence landscapes の空間には自然に $L^p$-norm が定義され Banach 空間になる。

$$\|\lambda\|_p := \left( \sum_{k=1}^{\infty} \|\lambda_k\|_p^p \right)^{\frac{1}{p}} \quad \text{for } 1 \leq p < +\infty\,,$$
$$\|\lambda\|_\infty := \operatorname{ess\,sup} \lambda = \|\lambda_1\|_\infty$$

Banach 空間に値を持つ確率変数に対しては、その積分[4]が定義できる場合があり、独立同分布に従う確率変数列に対する「大数の強法則」や「中心極限定理」が成立する条件が知られている。

[4]弱積分 (Pettis integral)

32 / 44

$L^p$-norm をよく見ると無限和、そして、無限区間の積分が関与しているから、計算機で計算しようと思うと結

構大変なのです。大変といいますか、無理なわけです、無限に足すということはできないですから。

特に実応用上重要な汎関数は

$$\lambda \mapsto \left( \sum_{k=1}^{K} \left\| \chi_{[-R,R]} \lambda_k \right\|_p^p \right)^{\frac{1}{p}}$$

である。ただし、$K$ は正の整数、$R$ は正の実数である。また、$\chi_{[-R,R]}$ は区間 $[-R, R]$ の指示関数である。
計算機で persistence landscape $\lambda$ の $L^p$-norm を計算するにあたっては、実数列 $\|\lambda_1\|_p^p, \|\lambda_2\|_p^p, \cdots$ 全てを足し合わせることはできず、事前に決め置いた $K$ 番目までの和を取って算出するより他なく、また各 $\lambda_k$ の $p$-norm も全範囲 $[-\infty, +\infty]$ で積分することはできず、事前に決め置いた $[-R, R]$ の範囲で積分して算出するより他ない。しかし、そのような簡便な方法で算出した $L^p$-norm に対しても、「大数の強法則」や「中心極限定理」が成立する条件が知られており、大変有意義なことである。

33 / 44

しかし、例えば有限和を取ってやって、また、積分区間も有限区間で積分してやる。これでも汎関数になっているのですけれども、これでも、実は大数の強法則や中心極限定理が成り立つということが分かります。だから、このような $L^p$-norm のいわば近似形のようなものに対しても、きちんと統計的に望ましい性質が成り立つということで、非常にありがたいことでありますというわけです。



34 / 44

最後に、数値実験の話をしようかと思います。

大阪取引所の銀先物価格に連動する東京証券取引所に上場されている上場投資信託[5]の価格データを用いて算出した persistence landscape を示す。

各営業日の「始値、終値、高値、安値」の 4 次元量を対数変換処理したデータを用いた。活用した期間は 2024 年 7 月から 2025 年 8 月までの 14 ヶ月間 (286 営業日) である。 この期間は植田ショック (2024 年 8 月 5 日) とトランプ関税ショック (2024 年 4 月 2 日) を含む。点群データとしては $\mathbb{R}^4$ 内の 286 個の点ということになる。

R パッケージ TDA を用いて、Vietoris-Rips 複体から 1-次までの persistent homology を算出し、persistence diagram と persistence landscape を描写した。

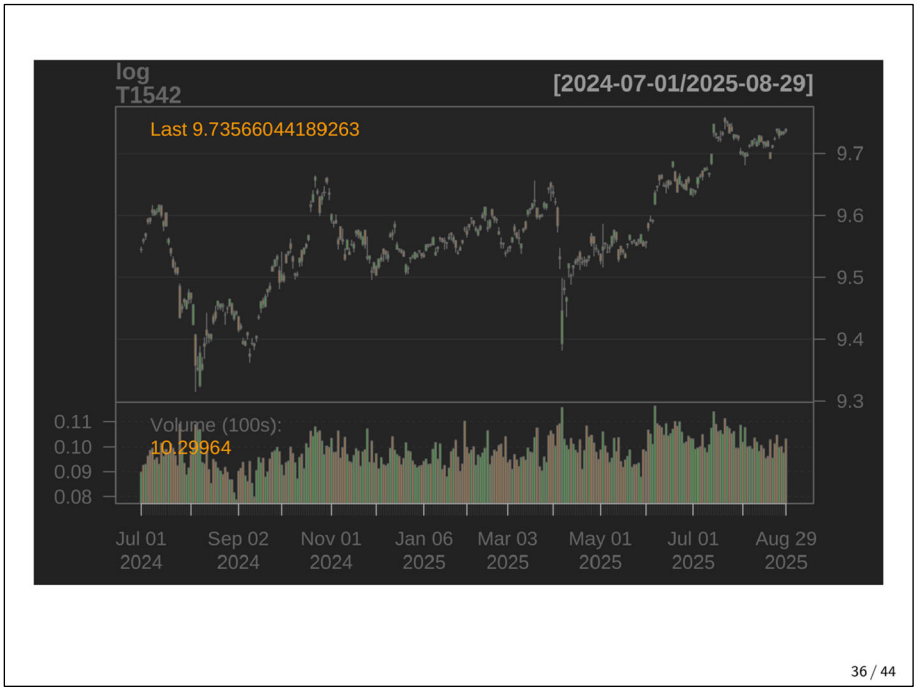
[5]三菱 UFJ 信託銀行『純銀上場信託（現物国内保管型)』証券コード: 1542

35 / 44

データとして今回使うものは、大阪取引所の銀先物価格に連動する、東京証券取引所に上場されている上場投資信託の価格データになります。少しややこしいですけれども、上場しているのは東京証券取引所です。けれども、連動指数は大阪取引所の銀先物価格ですということです。

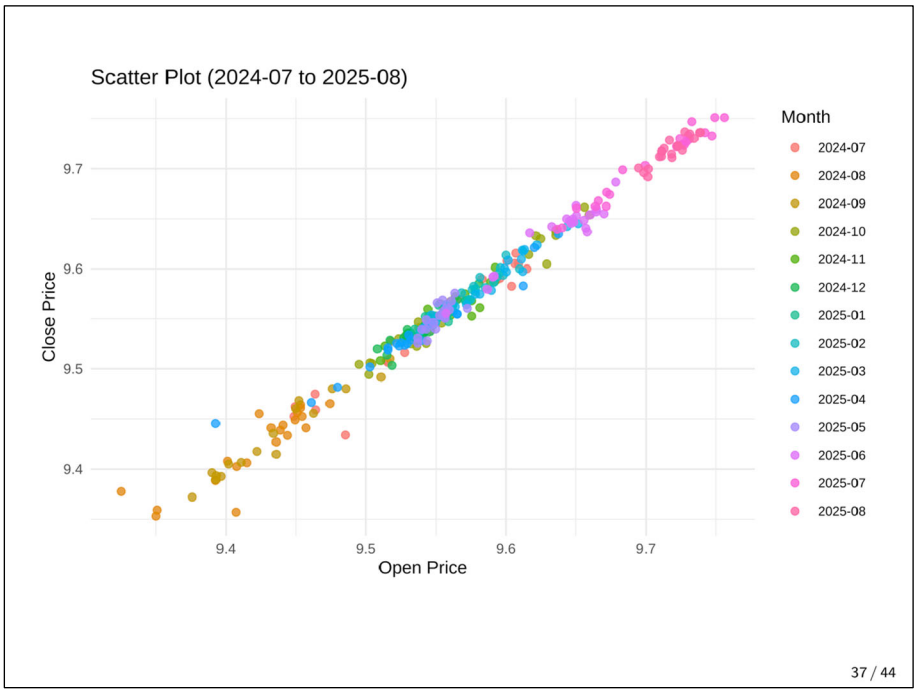
今回、これは金融データなのですけれども、各営業日の「始値、終値、高値、安値」の 4 次元量ですね。これは対数変換してやって、4 次元データとして取り扱っている。集計期間は 2024 年 7 月から 2025 年 8 月までの 14 ケ月間で、286 営業日でした。この期間は、植田ショックとトランプ関税ショックの 2 つのショックを含んでいます。点群のデータとしては 4 次元空間内の 286 個の点ということになります。

今回は、R パッケージの TDA を使っています。単体複体の計算方法としては、本日は Čech 複体と Vietoris-Rips 複体の両方を説明させていただいたのですけれども、Vietoris-Rips 複体の方を使っています。1-次までの Persistence Module を計算していて、Persistence Diagram と Persistence Landscapes を描いています。




こちらがまず、銀の価格で、これは対数表示している。この図だけを見ると、あまり変動しているように見えないのかもしれないのですけれども、銀は非常にボラティリティが高い金融資産で、これは実は激しく変動しています。これは対数表示しているから、それほど動いているように見えないかもしれないですけれども、もう 10%、20%、平気で上がったり、下がったりする金融資産です。

まず、ここで 1 回下がっているところがありますね。ここが植田ショック。ここで、もう 1 回大きく下がっているところがあり、ここがトランプの関税ショックになっています。このときは、本当に数日で 20%ぐらい価格が落ちて大変でした。集計期間は 8 月までしか取っていないのですけれども、ご存じのとおり、9 月は貴金属が暴騰していますので、今もこのように来ています。

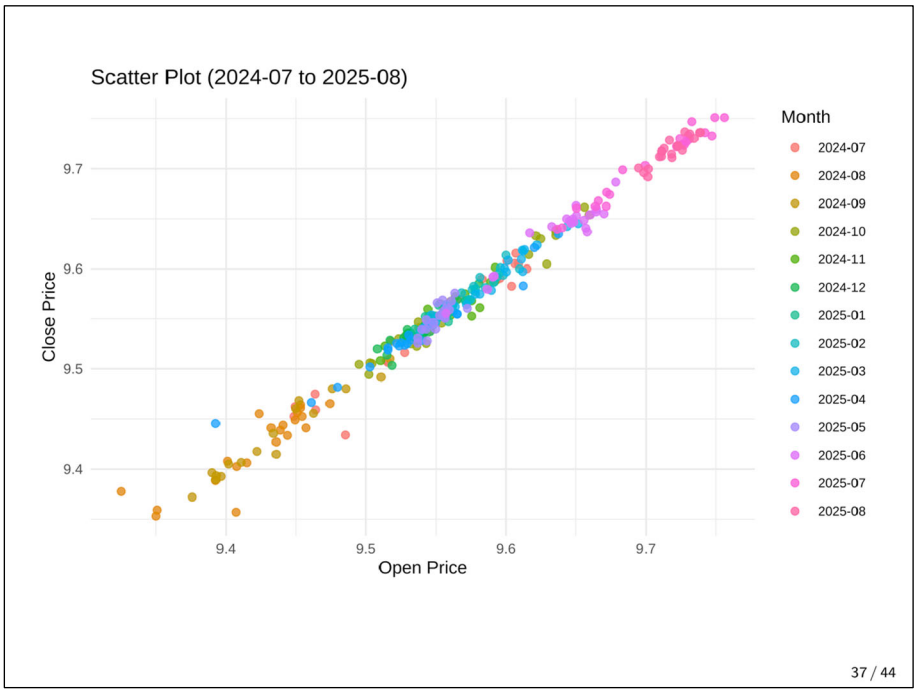



こちらは、元々4 次元のデータなのですけれども、4 次元のうち始値と終値で 2 次元プロットしたもので

す。基本的には、始値と終値と高値と安値、連動し合っているので、4 次元データとはいっても基本的には 1 次元データなのです。でも少し、やはり始値の方が終値より低かった日、そうではない逆の日などがあるわけで、少し 45 度線から離れる点もいくつかある。

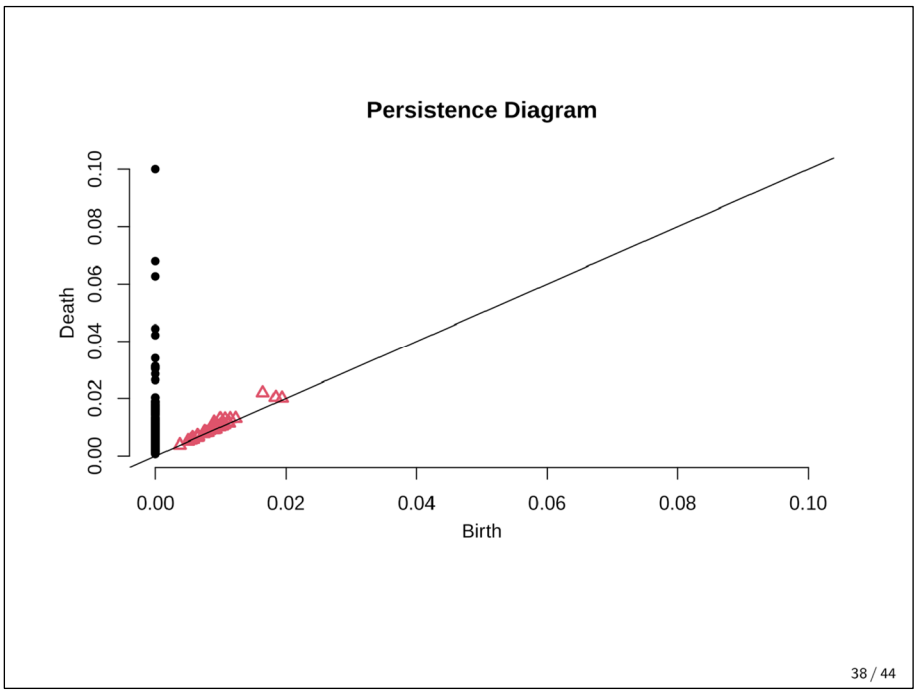
この色分けは、月を表していて、


もう 1 回銀の価格変動を見ると、

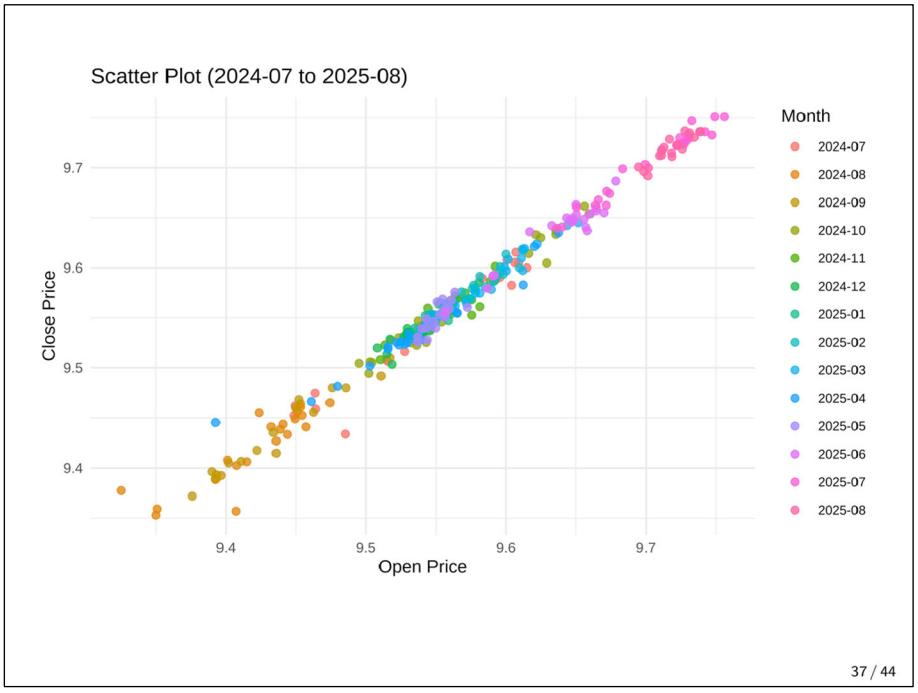



最初はこのあたりなので、最初このあたりにいて、植田ショックがあって、ここに飛んでいるということですね。そのあと価格が上がって、このあたりに来ていて、トランプショックのとき、この青色の丸、ここの青色の丸で 3 つぐらい飛んでしまっているところがあると思うのですけれども、ここがトランプショックのときで、銀の価格が暴落したところになります。そのあとは、貴金属は銀に限らず上がり続けて、今も上が

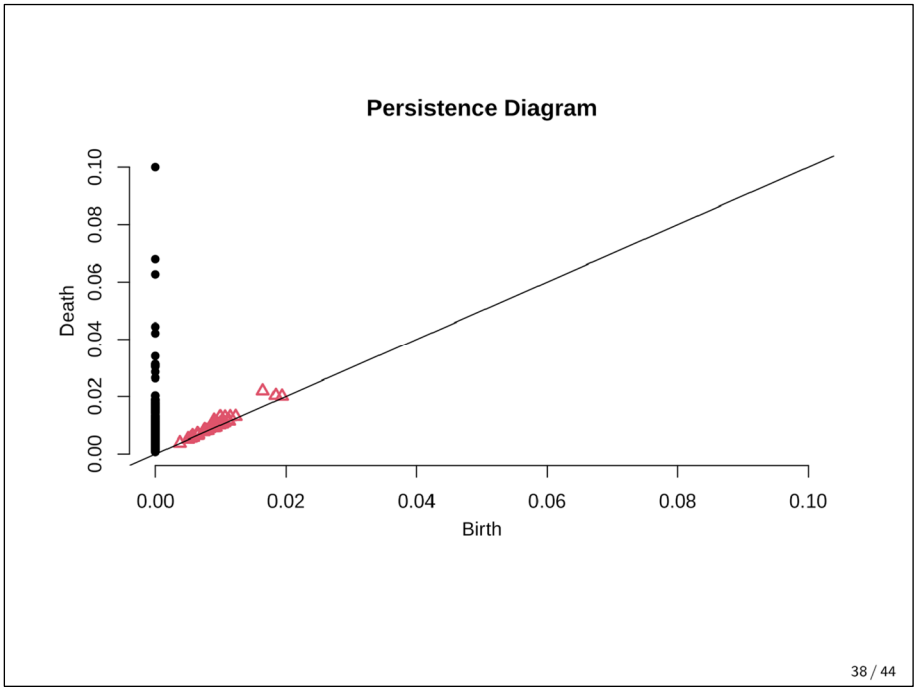
り続けているというような状況です。ようやく、先月末ぐらいに少し調整が入って、少し落ち着いたかというところです。



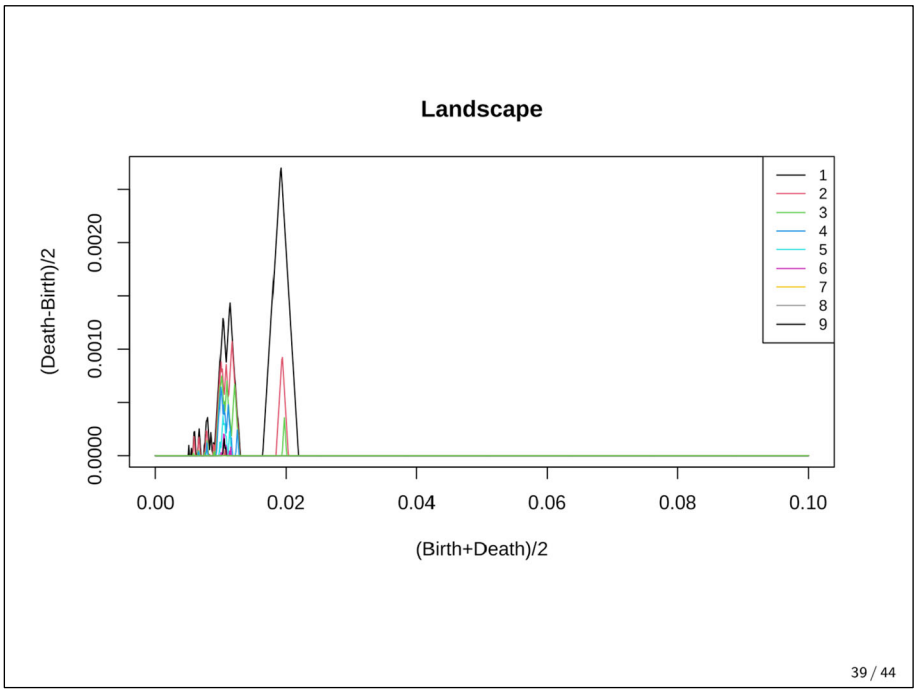

こちらがその Persistence Diagram です。




4 次元評価の中の 286 個の点に対して、




Persistence Diagram を計算するのですけれども、この黒い丸の方が 0 次元です。この三角形の方が 1 次元で、この 45 度線から離れているほど顕著で、例えばこのようなところに飛んでいるものが見えます。




こちらが、Persistence Landscape の方です。Persistence Landscape というものは、実数直線上で定義された実数値関数列なのですけれども、1 番上が 1 つ目の $\lambda_1$ です。次の赤い色のところが $\lambda_2$。その次が $\lambda_3$ という感じで、このようなギザギザを計算して何の意味があるのかと思っている方もいらっしゃるかもしれないですけれども、実はこれが結構、元々のデータの要約量になっていて、いろいろな使い道があるのです。

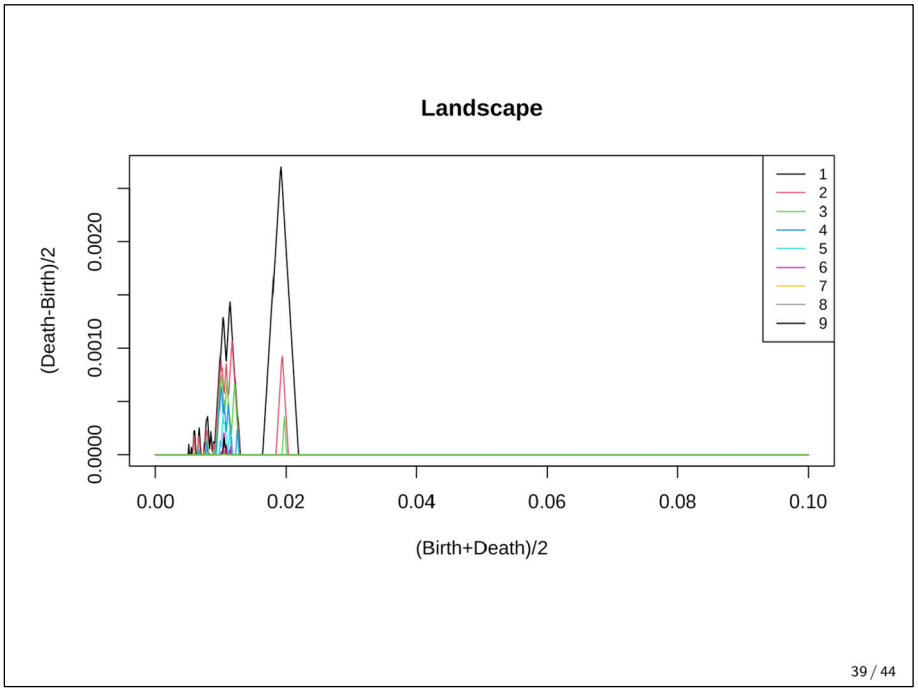
同様にして、東京証券取引所に上場されている上場投資信託 11 銘柄に関して、Vietoris-Rips 複体を構成し、persistence landscapes に対して $L^2$-norm に基づく階層的クラスタリングを実施した。

データの集計方法は同一（対数処理した 4 次元データ、集計期間は 2024 年 7 月から 2025 年 8 月まで）である。

階層的クラスタリングには R パッケージ stats の hclust 関数 (method = "ward.D2") を用いた。

40 / 44

その話を少し、これからしてみましょう。今、銀を見たのですけれども、銀以外も見てみましょう。東京証券取引所に上場されている上場投資信託 11 銘柄に対して、同じことをしました。Vietoris-Rips 複体を構成して、Persistence Landscapes を計算して、さらに $L^2$-norm に基づく階層的クラスタリングを実施した。$L^2$-norm といっても、




39 / 44

このように Landscape を計算してやると、それによって、Landscape 同士に対して $L^2$-距離というものを計算することができるから、それに基づいて、階層的クラスタリングを実施することができるというわけです。

同様にして、東京証券取引所に上場されている上場投資信託 11 銘柄に関して、Vietoris-Rips 複体を構成し、persistence landscapes に対して $L^2$-norm に基づく階層的クラスタリングを実施した。

データの集計方法は同一（対数処理した 4 次元データ、集計期間は 2024 年 7 月から 2025 年 8 月まで）である。

階層的クラスタリングには R パッケージ stats の hclust 関数 (method = “ward.D2”) を用いた。

40 / 44

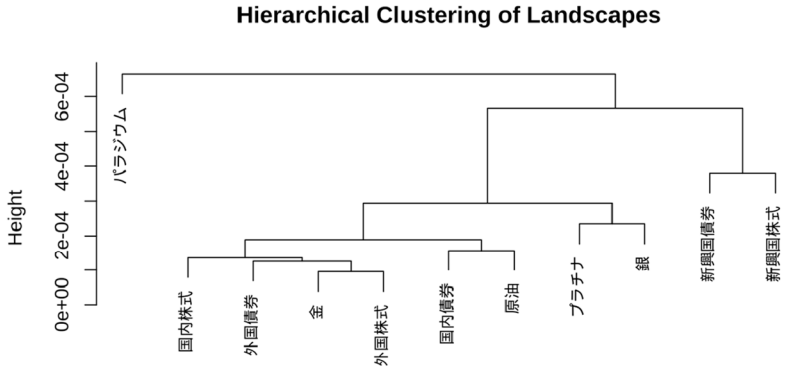
データの集計方法は全部同じで、やはり対数処理をした 4 次元データで、集計期間は同じというわけです。なぜ、対数処理をしておくかというと、スケールが違うと一般的に TDA は難しいのです。ある軸を 500 倍などにすると、前の空間では円だと思っていたものが、スケールを変えてしまうと長円といいますか、真ん丸ではなくなってしまいます。だから、スケール依存性があるのですけれども、対数処理をしておくと、定数倍ファクターは平行移動にできます。だから、例えば金と銀があって、金と銀の価格の比率は、今 80 倍ぐらいなのですけれども、もし、その比率が変わらないのだとするならば、対数を取ってやると、定数倍ファクターというものは定数平行移動になりますから、その定数平行移動をしても、Persistence Landscape は変わらないということは分かると思う。幾何的な情報ですから、平行移動しても変わらないですからね。だから、もし、定数倍ファクターしかないと思っているのであれば、そのファクターは対数処理しておくと消えてなくなるので、対数を取っておくことは大事です。階層的クラスタリングには、R パッケージの標準的なものを使っています。

| 証券コード | 連動指標 |
|---|---|
| 1540 | 大阪取引所の金先物価格 |
| 1541 | 大阪取引所のプラチナ先物価格 |
| 1542 | 大阪取引所の銀先物価格 |
| 1543 | 大阪取引所のパラジウム先物価格 |
| 2510 | NOMURA-BPI 総合 |
| 2511 | FTSE 世界国債インデックス（除く日本） |
| 2519 | J.P. Morgan - EMBI Plus |
| 1306 | TOPIX（配当込み） |
| 2513 | MSCI – KOKUSAI 指数 |
| 2520 | MSCI - Emerging Markets 指数 |
| 1699 | NOMURA 原油ロングインデックス |

全銘柄、円建、為替ヘッジ無し。

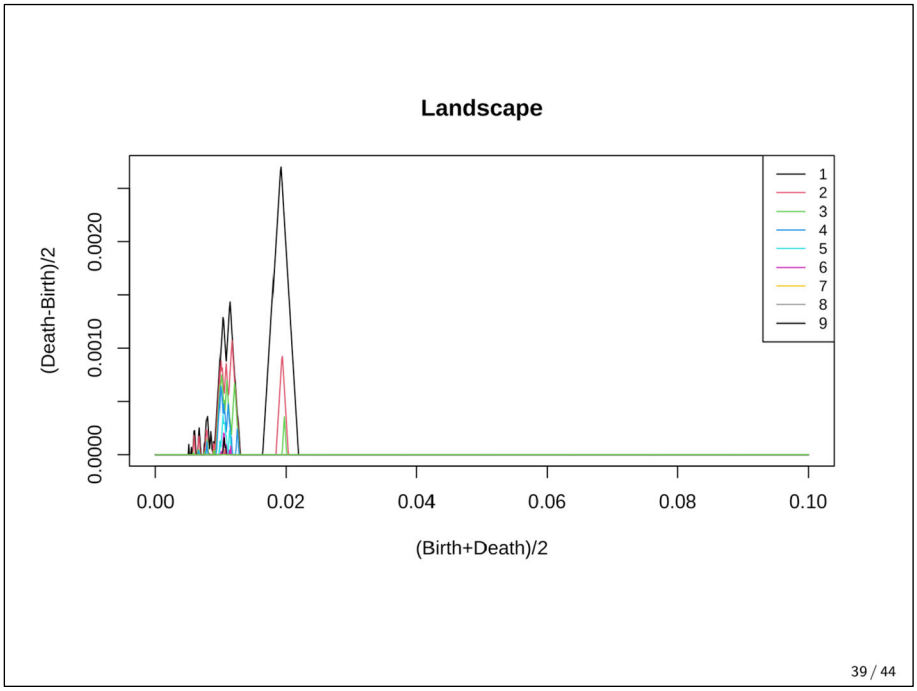
41 / 44

11 銘柄というものは、今回、次のようなもので、上の 4 つが、いわゆる貴金属です。人気があるもので、一番有名なものが、金、プラチナ、銀、パラジウムです。全部、先物価格です。次の 3 つが債券です。次の 3 つが株式で、最後が原油。全部、これは東京で上場されているものなので、円表示です。原資産がドルであろうが、関係ないです。東京証券取引所は全部円で売買されているので、全部円表示で、為替ヘッジは無いです。

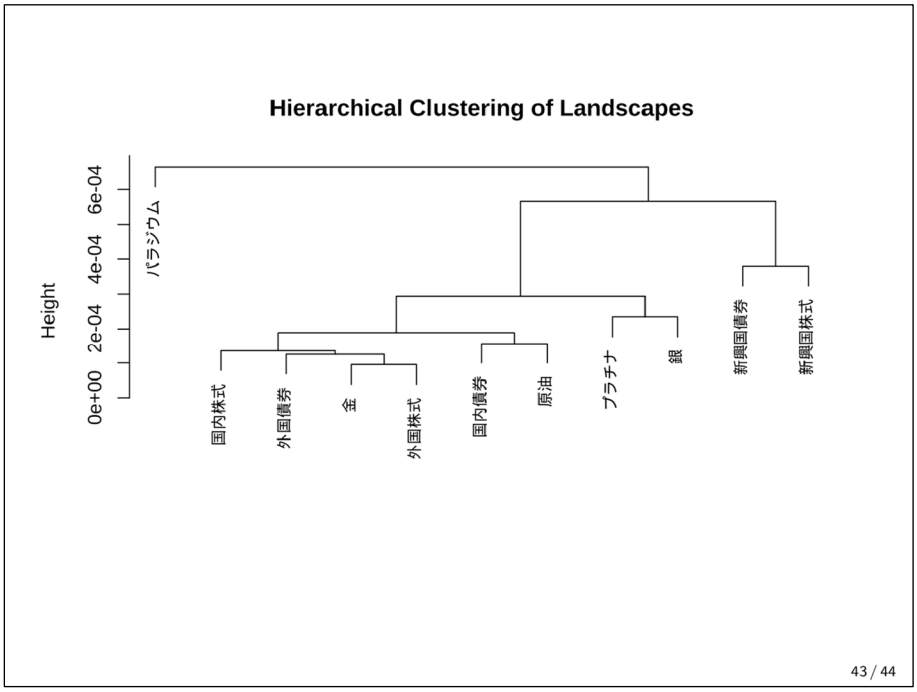



43 / 44

先に結果をお見せします。このような感じで、11 個の金融資産のクラスタリングをしたものです。これを見て、よくできていると思う人もいれば、これがきちんと、いいクラスタリングなのか分からないという人もいるかもしれないですけれども、私個人としては結構よくできているかと思っています。

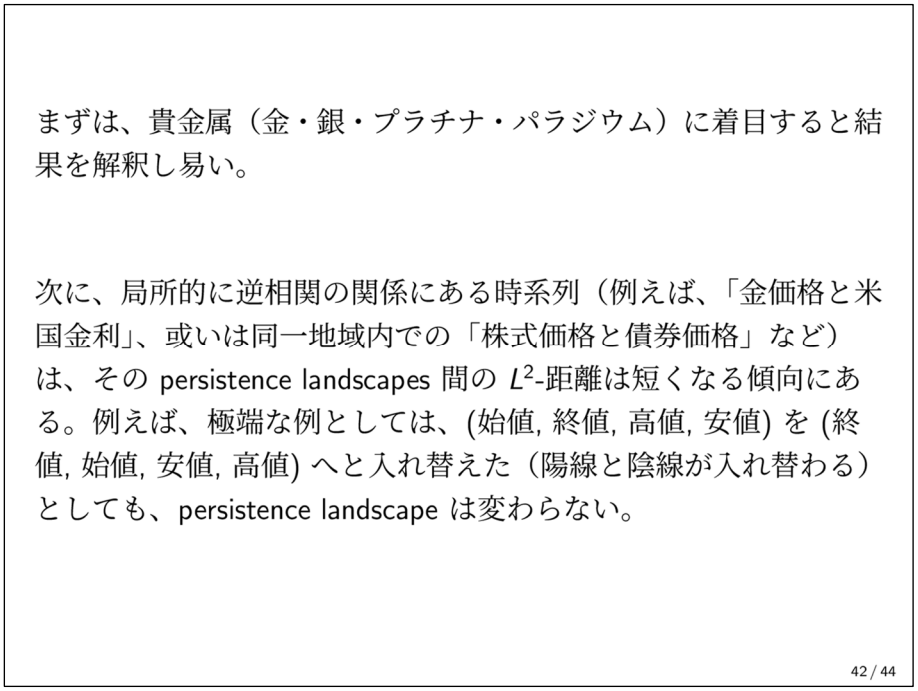



しかも、クラスタリングを使ったものは、このようなただのギザギザなわけですね。こういった、ただのギザギザが、いかほどの情報を持っているのかということは、まだ、このギザギザを見ても分からないと思うのですけれども、このようなギザギザでも、それなりの情報を持っていて、きちんとクラスタリングするといい感じに分かれている様子が見える。

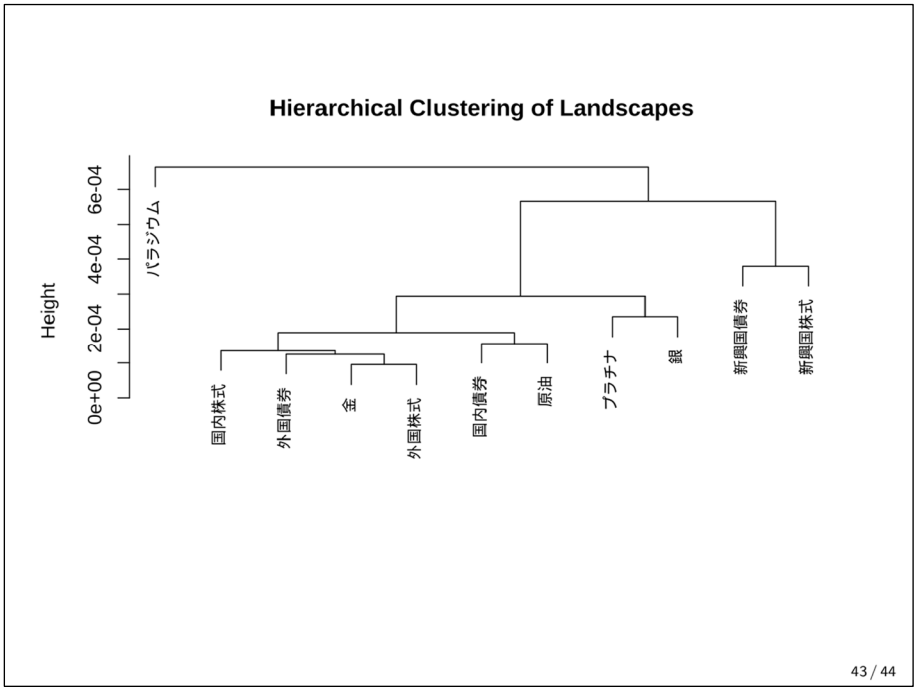



このようなクラスタリングがよくできているか、よくできていないかを見るときに、まず、貴金属に着目してほしい。貴金属というものは、金と銀とプラチナとパラジウムのことで、この位置関係ですね。よさそうに見える。金というものは価格の王様という感じで、これがすべてのモノの値段をつかさどっているというように考えてもらうといいかと思います。銀とプラチナは兄弟のようなもので、金とよく似た性質もあるが、一方で、工業需要で価格が決まるような側面もあります。よく銀とプラチナは金の価格を後追いするな

どといい、やはり連動するのですけれども、全く同じように動くわけでもないです。パラジウムは確かに仲間はずれで、これはどのような理由で価格が決定されていくのか、よく分からない貴金属です。基本的には工業需要で価格が決まる金属です。だから、貴金属の位置関係は非常によさそうに見える。




次に、このような観点を知っておくと分かりやすいかと思っていて、ミクロに逆相関の関係にある時系列は、L2-距離は短くなります。極端な例としては、始値・終値・高値・安値を、終値・始値・安値・高値に入れ替えるようなことを考えてみましょう。要するに、陽線と陰線が入れ替わるようなケースです。例えば株式と債券のように、マクロがどのような関係なのかよく分からないけれども、ミクロには株価が上がるときに債券が下がるような、局所的な逆相関の関係がある。そのようなものは、実は、幾何的な情報は何も変わらない。4 次元の軸を入れ替えているだけだから、幾何的な情報は何も変わらなくて、Persistence Module は変わらないので、当然 Persistence Landscape も変わらなくて、$L^2$-距離も、完全にこのように入れ替わるなどすると、$L^2$-距離はゼロになるわけです。だから、局所的に逆相関にある時系列は近づきます。




そういったところが、こういったグラフでも見えていて、例えば新興国債券と新興国株式など、同じグループに入っているところが見えます。

こちらのクラスターも分かりやすい。金は円建てだとどのように価格が決まるかというと、実はドルの金利に一番影響を受けます。ドルの金利が下がると相対的に金の魅力が上がるので、まず、ドル建てで見て金の価格が上がりますね。それだと、ドル建てで見ているだけなのですけれども、為替の変動を込みで見ても、やはりドルの金利が下がると、円建てでも見ても金の価格は上がります。だから、円建てで見ても、ドルの金利が一番効いているわけです。

それを言うと、外国債券も同じです。外国債券というものは、要するに米国の債券のことで、ドル建ての債券ですから、ほとんどが。つまり、米国金利が上がると債券が下がって、金も、外国債券も、基本的には米ドルの金利に振り回される。外国株式も同じです。長期的にどのような影響を受けるかは非常に難しいのだけれども、短期的には外国の債券も、外国の株式も、やはり動く方向は逆かもしれないけれども、米国のドルの金利に非常に影響を受けているので、このあたりが近づくということは自然ですね。

国内の株式はほぼ米国の株式に連動するので、このあたりも同じグループに入っているという様子が見えます。国内債券は、このグループに入らないです。国内債券は米ドルの金利はほぼ関係なくて、円建てで見られますから、米ドルの金利ほど関係しなくて、これは日本銀行が金利をどう設定するかで決まる。日米の中央銀行は、基本的には独立に金利を設定しているので、国内債券の方は米国の中央銀行の金利にはあまり振り回されなくて、少し距離を取っている様子が見えます。

Persistence Landscapes の性質を知っていれば、このようなクラスタリングの結果になるということは、比較的自然な結果なのかと思っているというわけで、そうですね、はい。

私からの発表は以上になります。

では、少し早いですけれども、汪の方から次の応用の話ですね。私の方は、割と数学的な話の基本的な事項をまとめておいたので、もう少し金融時系列への応用に関して、説明していただこうかと思います。

# 位相的データ解析入門
# Persistence Landscape の応用

## [Gidea and Katz 2018] 再現

汪 沁亮

2025/11/07

**【汪】** 皆さん、こんにちは。汪と申します。

本日は、小田さんにお話しいただいた定義と性質を踏まえ、応用の観点から、金融時系列データに対してトポロジカルデータ解析(TDA)、特に Persistence Landscape を使った分析の例をご紹介します。これは Gidea さんと Katz さんが 2018 年に発表した研究の再現です。

## 概要 - [Gidea and Katz 2018] 再現

**Gidea, M., & Katz, Y. (2018). Topological data analysis of financial time series: Landscapes of crashes. Physica A: Statistical mechanics and its applications, 491, 820-834.**

**原論文では、TDA を用いた金融時系列の分析により、金融危機の事前察知が可能であることを示唆している。**

原論文に従い、米国の主要株価指数である S&P500、DJIA、NASDAQ、Russell 2000 の日次対数リターンの 4 次元データを用い、ウィンドウ（連続した部分時系列）をスライドさせながら、Vietoris-Rips 複体から 1 次までの persistent homology を算出し、その persistence landscape の L1-norm 及び L2-norm を 1 次元時系列データとして描写した。

原論文では 1987/12/23 ~ 2016/12/08 まで（7301 営業日）のデータ（2000 年ドットコムバブルと 2008 年金融危機を含む）が用いられたが、今発表では 1995/01/01 ~ 2025/10/01 まで (7737 営業日) のデータ（2020 年コロナショックと 2025 年トランプ関税ショックを含む）を用いた。ウィンドウ幅は原論文と同じ 50 営業日を用いた。

Vietoris-Rips 複体の算出には R パッケージ **TDA**（library = "GUDHI"）を用いた。

この研究では、株価の変動を単なる価格の推移ではなく、データの形や構造の観点から分析しています。

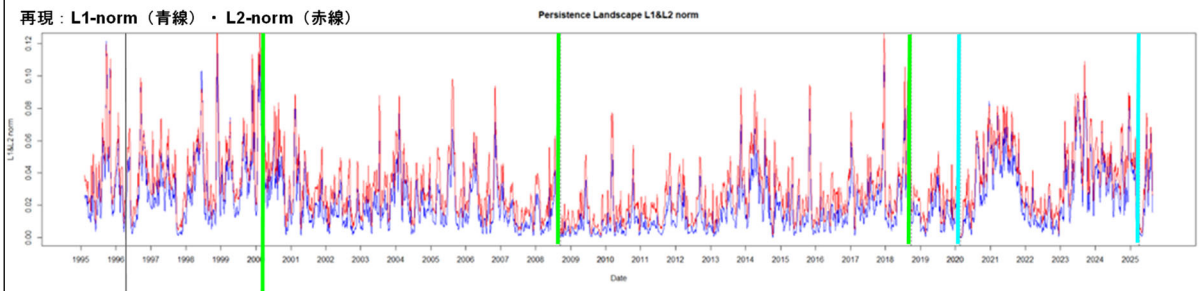
こちらに書いてある S&P500 の日次対数リターンを使い、50 日ごとの部分データを解析し（50 営業日をスライドウィンドウとして)、Vietoris-Rips 複体を構築して persistent homology を計算しました。これにより、データの中に隠れた「ループ」や「穴」といった構造を捉えます。

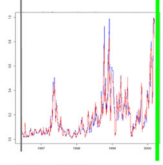
得られた結果を Persistence Landscape として可視化し、$L^1$・$L^2$-norm で時系列化することで、市場構造の

変化を追跡できます。norm が上昇するときは、市場の構造が複雑化している、すなわち変動が大きくなっている可能性を示します。

## Persistence Landscape の L1-norm, L2-norm の再現

再現：L1-norm（青線）・L2-norm（赤線）



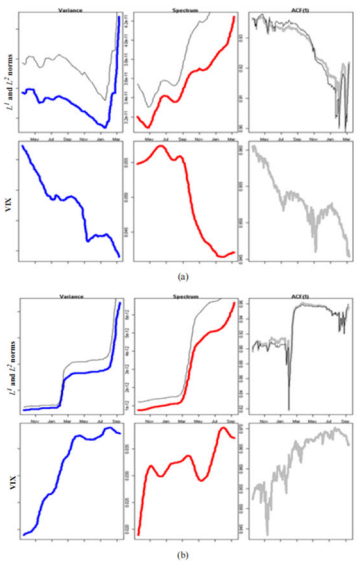
"Figure 9" in [Gidea and Katz 2018]

原論文で指摘されているように、2000 年インターネットバブルと 2008 年金融危機の直前で persistence landscape の Lp-norm の上昇が認められる。また、対中関税第３弾が発動された 2018 年 9 月 24 日の直前にも同様の上昇が認められた。

一方で、2020 年コロナショックと 2025 年トランプ関税ショックでは明白な上昇を認めることができなかった。

実際の分析では、左側 2000 年の IT バブル崩壊前や 2008 年のリーマンショック前に norm が明確に上昇していました。下図は原論文の図です。再現結果とは完全には一致していませんが、概ね同様の形になりました。また、2018 年の対中関税発動前にも同様の傾向が見られました。一方、2020 年のコロナショックや 2025 年のトランプ関税ショックでは同じような上昇は確認できませんでした。こちらの水色の線です。

"Figure 11" in [Gidea and Katz 2018]



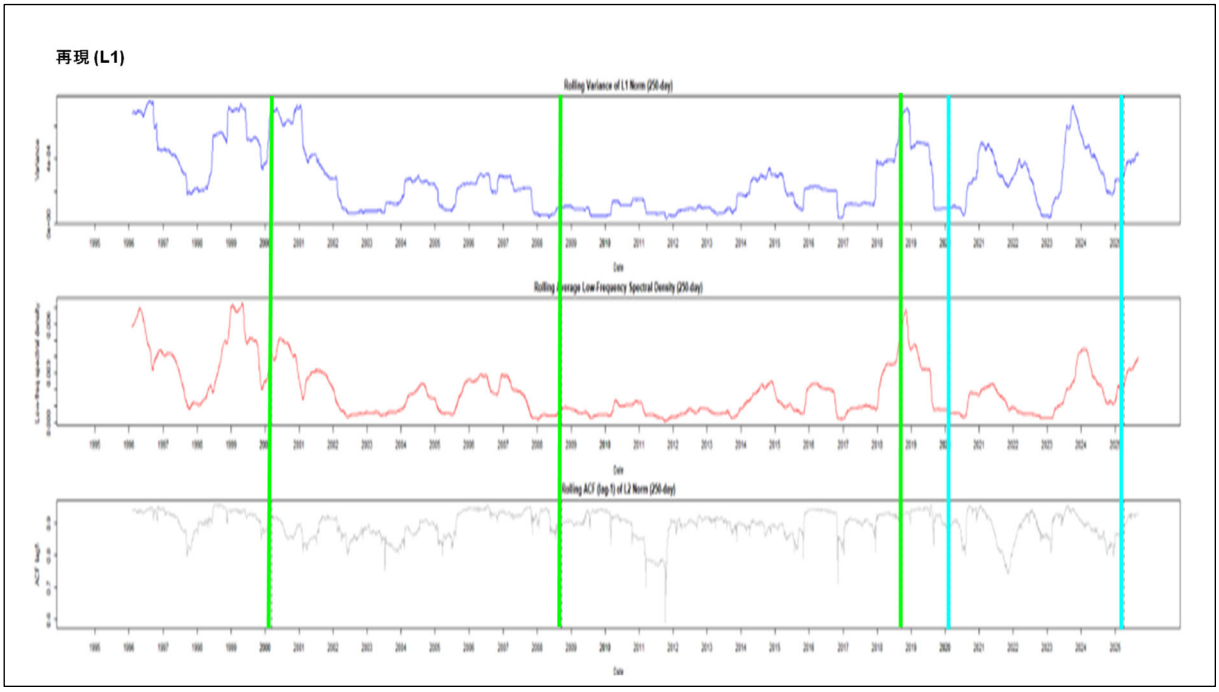

原論文では、persistence landscape の Lp-norm に関して、(1) 移動平均の分散 (2) スペクトル密度関数の低周波領域のスペクトル密度の平均 (3) 一次自己相関 (ACF) に言及がある。

上図上段は 2000 年インターネットバブル時の、下図上段は2008 年金融危機時の、(1), (2), (3) が描写がされている。

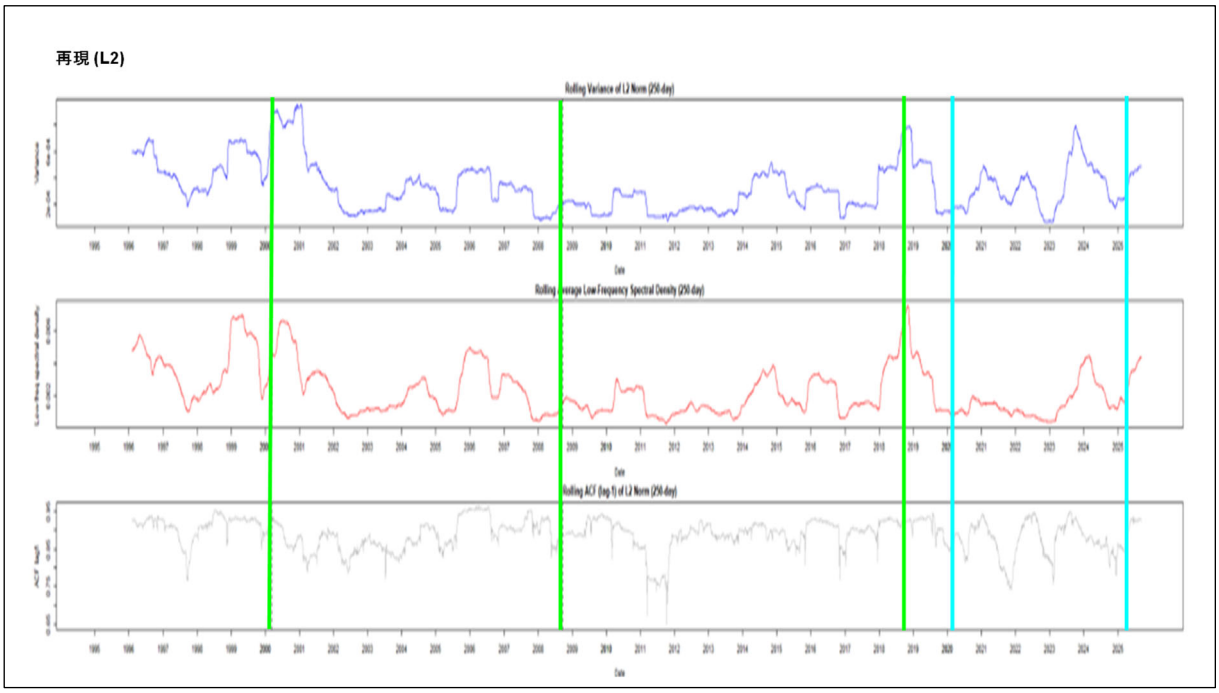
原論文では、バブル崩壊・金融危機の直前 250 日で (1), (2) の上昇が認められるとしている。

本発表では、集計期間を 1995/01/01 ~ 2025/10/01 に拡大し、(1), (2), (3) を再集計し、分析を試みた。

加えて、移動平均の分散や低周波スペクトル密度、1 次自己相関といった統計指標も比較したところ、




2000 年、2008 年、2018 年の危機前には上昇が見られた一方、2020 年や 2025 年の危機前には明確な上昇は確認されませんでした。




こちらは、$L^2$ の再現です。同じ結果になっています。

まとめると、Persistence Landscape は、市場の構造変化を早期に把握する可能性を持つ有用なツールです。ただし、すべてのショックで norm が上がるわけではなく、市場の性質やショックの種類によって結果は異なることも分かります。

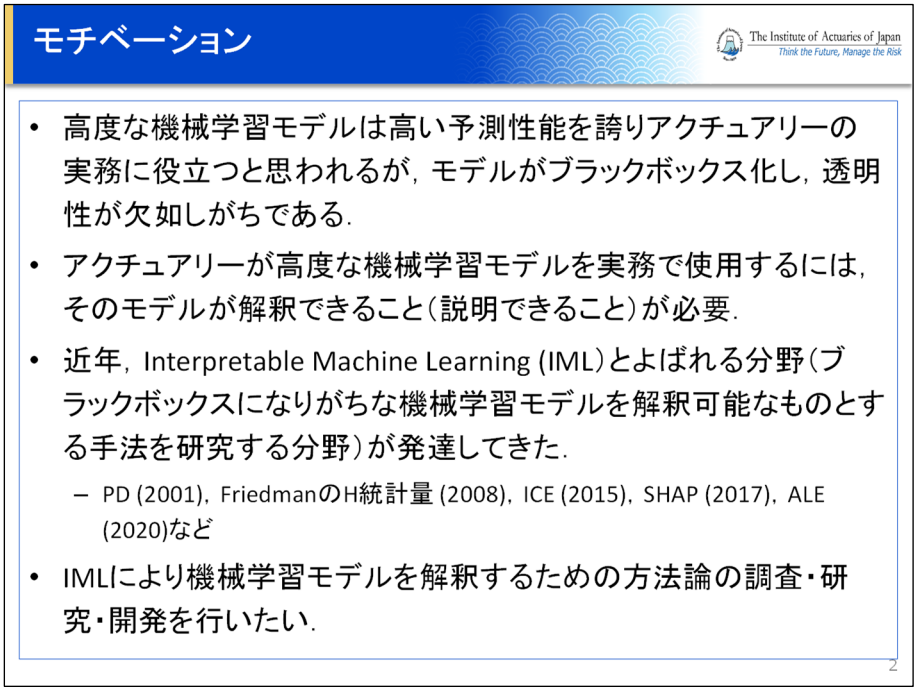
以上が今回の再現分析の概要です。ご清聴ありがとうございました。

**【司会】** それでは、前半の質疑応答に入ります。会場にいる方で、質問のある方は挙手をお願いします。

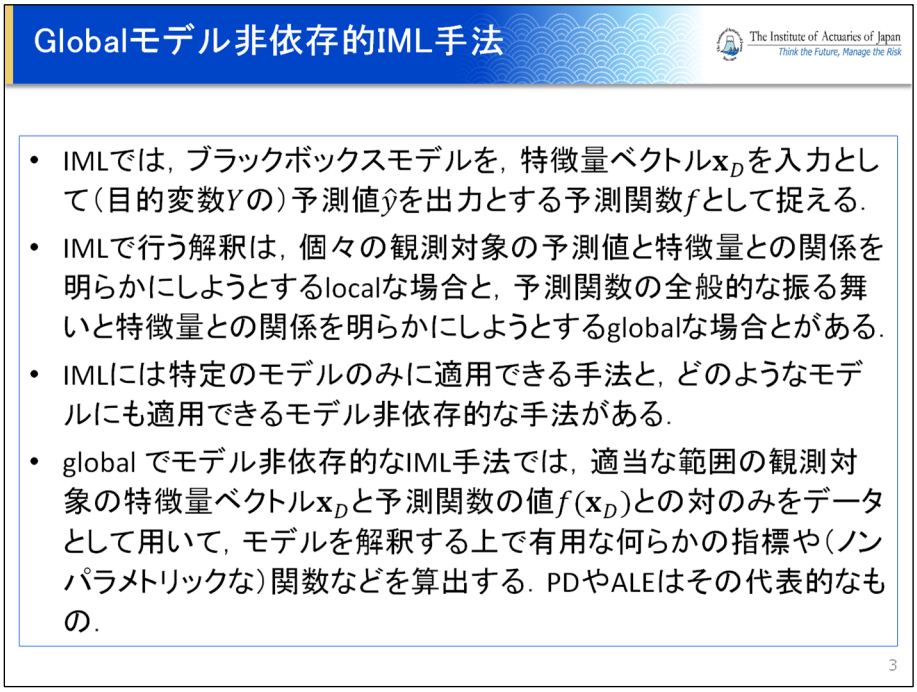
Slido には質問がないようですので、後半に移ります。小田さん、汪さん、ありがとうございます。

それでは、松森さん、浅芝さん、よろしくお願いします。


**【松森】**　では、後半の発表を始めさせていただきます。後半の前半、すみません、ややこしいのですけれども、担当させていただきます、SOMPO リスクマネジメントの松森と申します。よろしくお願いいたします。後半の内容につきましては、IML ですね。機械学習の結果できた複雑なモデルを、解釈できるようにする手法です。岩沢さんが発案された、MID という方法がございます。データサイエンス関連基礎調査部会の IML チームというものがありまして、そちらの浅芝さん、上妻さんがその手法を、実際に R パッケージを開発して、CRAN に登録されたという活動がございます。後半は、そのパッケージのご紹介を浅芝さんからいただきます。前半では、その MID という手法がそもそもどのようなものなのかというところと、なぜ、そのような手法が提案されるに至ったのかというようなところを、私の方からご紹介させていただきます。




はい。モチベーションとしましては、よくいわれるように、機械学習モデルはブラックボックス化しがちである。アクチュアリーも、もちろん機械学習のモデルを使いたいのですけれども、やはり説明責任がございます。解釈できるということが非常に大事になってくるので、この IML を研究したいというところで、まず、モチベーションとして大きいものがございます。



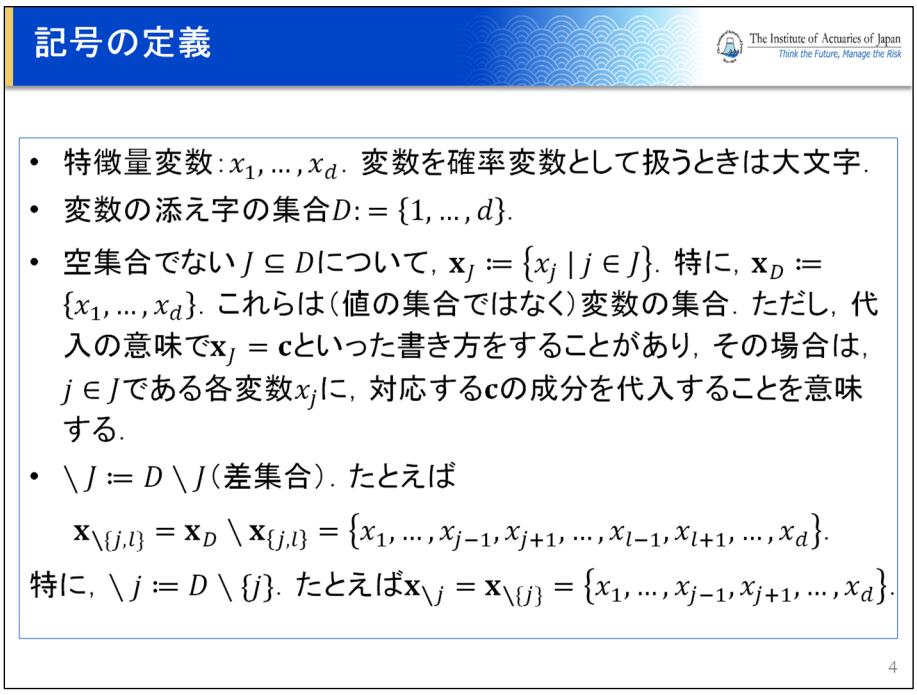

具体的には、岩沢さんが扱っている IML 手法というものは、いろいろと種類があるのですけれども、global でモデル非依存的な手法を扱っております。この IML では、ブラックボックスモデルを単純に関数と捉えます。つまり、特徴量を放り込んだら、予測値が返ってくる関数です。定義域が特徴量の空間であって、レンジ、値域はシンプルに実数が返ってくる。それを関数と捉えます。

まず、global というものと local というものがあるのですけれども、local は個々の予測に対して解釈を

行うというものです。一方、global は予測関数、ブラックボックスのモデルが全体として、どのようにふるまっているのかというところを明らかにしようというものです。あとは、そのモデルがどうやって作られたかというところに依存して、例えばランダムフォレストで作られたモデルにしか使えませんというようなものもあるのですけれども、いや、どのようなモデルで作られたものであっても、インプットとアウトプットの関係さえあれば、解釈できるというモデル非依存的な手法があります。この MID はグローバルな、モデル非依存的な手法として研究されております。

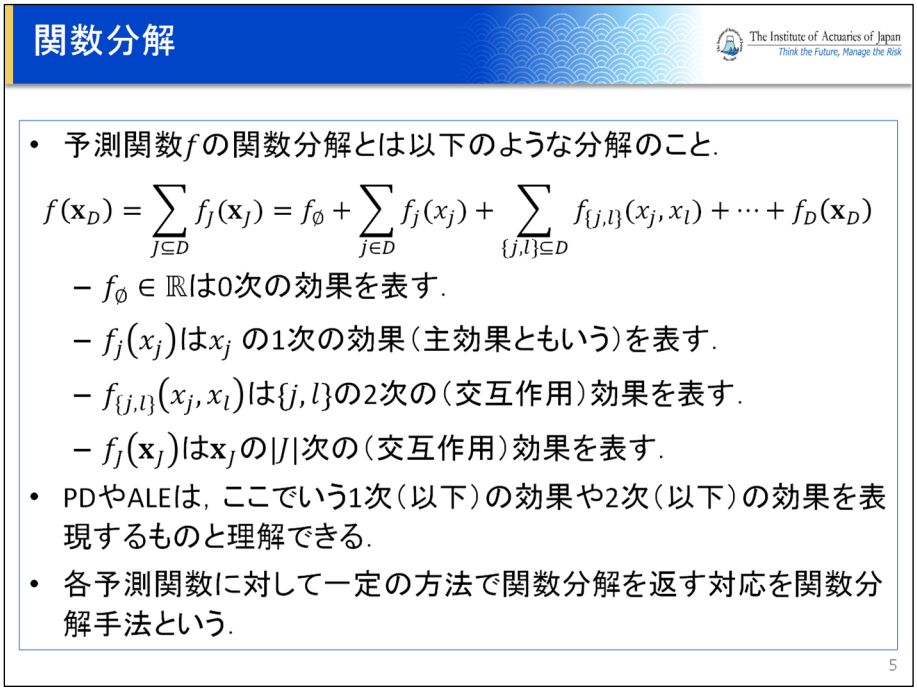
ここで、PD や ALE がその代表的なものというところで、1 番下に書いてございます。IML 手法にもいろいろなものがあるのですけれども、global でモデル非依存的な手法としてこのようなものがございます。具体的な定義などは、後でご説明いたします。

ここで 1 つ言いたいことは、機械学習の文脈のお話をさせていただくのですけれども、通常よくあるような、どのようにモデルを作るのかという話は、今日は一切いたしません。何か関数が与えられました。よく分からない関数が与えられましたというときに、そのふるまいを知りたい。そのためにどうすればいいかという話にフォーカスさせていただきます。




ここから準備が続くのですけれども、記号の定義でございます。まず、特徴量の変数は $x_1$ から $x_d$ の d 個あるといたします。確率変数と思うときは大文字で記載します。変数の添え字は 1 から d まであるわけなのですけれども、D というものをこの 1 から d の集合といたします。その D の部分集合は J でよく書きます。ボールド $\mathbf{x}$ の下に J と書いたら、その部分集合 J に入っている添え字の変数の集まりという書き方をいたします。たまに、この $\mathbf{x}_J=\mathbf{c}$ など、今日は出てこないかもしれないのですけれども、そのような書き方をすることがありまして、これは各 J の元 j である変数に対応する変数 $x_j$ に、$\mathbf{c}$ の j 成分を代入するという意味で使います。

この＼J というものは、D から J を引いた差集合のことを表します。この J が特に 1 点集合であるときは、この集合を表す括弧を省いて、＼j と単純に記載いたします。こちらは ALE という、先ほど先行手法が出てまいりましたけれども、そちらを導入している論文にならった記法ということになっております。




global でモデル非依存的な IML 手法を研究していきたいのですけれども、その手法として、関数分解という手段を採用しております。関数分解とは何かといいますと、まず、関数が何か与えられましたと。$f(\mathbf{x}_D)$ですね。引数は 1 から d まですべての変数を引数に取る関数 f が与えられたというときに、それを、この$f_\phi$というものは定数項です。その次に 1 点集合を全部集めてきて、その 1 つだけの変数を引数に取るような 1 変数関数の和と表して、その次に 2 変数関数の和と表して、ばっと 1 次の項、2 次の項、3 次の項というようにして、全部足すと元の関数が復元されるようなもののことを、関数分解と呼びます。

分解としては、別に何でもいいから分解してしまえばいいのですけれども、解釈という意味では、この$f_\phi$という定数項は 0 次の効果、要するに期待値的なものです。$f_j(x_j)$というものは、1 変数関数はその変数による主効果で、j、l という 2 点集合からなる 2 変数関数は、$x_j$と $x_l$の 2 次の交互作用を表す項であるというように解釈いたします。

PD や ALE は明示的に関数分解を返すものではないのですけれども、これを使って関数分解をすることもできます、後で触れるのですけれども。いわゆる PD や ALE というものは、その関数分解における 1 次の部分や 2 次の部分を返していると理解することができます。

言葉の問題ですけれども、関数を何でもいいから分解することを、関数分解というのですけれども、関数分解手法と言ったときは、その関数に対して関数分解を対応させる作用素のことを、関数分解手法ということにいたします。ですから、global でモデル非依存的な IML 手法として、適切な関数分解手法を見つけたいというモチベーションになってまいります。

## PD(Partial Dependence)

The Institute of Actuaries of Japan
Think the Future, Manage the Risk

- 1次元PDの概念は，$f_\emptyset + f_j(x_j)$として$\mathrm{PD}_j(x_j) := \mathbb{E}[f(x_j, \mathbf{X}_{\setminus j})]$を提案したものと理解できる．
- 実用上の1次元PDは $\mathrm{PD}_j(x_j)$を推定した結果の関数$\widehat{\mathrm{PD}}_j(x_j)$であり，推定に使う$\mathbf{X}_{\setminus j}$のデータを$\mathbf{x}_{\setminus j,1}, \dots, \mathbf{x}_{\setminus j,n}$とすれば，$\widehat{\mathrm{PD}}_j(x_j) = \frac{1}{n}\sum_{i=1}^{n} f(x_j, \mathbf{x}_{\setminus j,i})$．
- 高次のPDも同様に定義できて，それを基にした関数分解手法も定義できる．
- 簡単のため，以後この関数分解手法も単にPDと呼ぶ．

6

ここから、PD と ALE の簡単な導入をさせていただきます。いわゆる 1 次元の PD というものは、先ほど 1 次の項と言ってしまったのですけれども、その 1 次以下の項ですね。定数項も含めて、この$f_\emptyset$+$f_j(x_j)$というところをここに定義する。少しややこしいのですけれども、f から $x_j$ を固定して j 以外の部分を確率変数と思って期待値を取ったものというように理解できます。実際に計算するときは、下のように計算するのですけれども、この定義は非常に自然ですので、1 変数で、今、書いていますけれども、j を J にすれば高次の PD というものも定義できまして、それを基に関数分解をすることができます。

## ALE(Accumulated Local Effects)

The Institute of Actuaries of Japan
Think the Future, Manage the Risk

- 1次元ALEの概念は，$f_j(x_j)$として次の$f_{j,\mathrm{ALE}}(x_j)$を提案したもの．

$$f_{j,\mathrm{ALE}}(x_j) := \int_{x_{\min,j}}^{x_j} \mathbb{E}\left[\frac{\partial f}{\partial x_j}\middle| X_j = z_j\right] dz_j - c_j$$

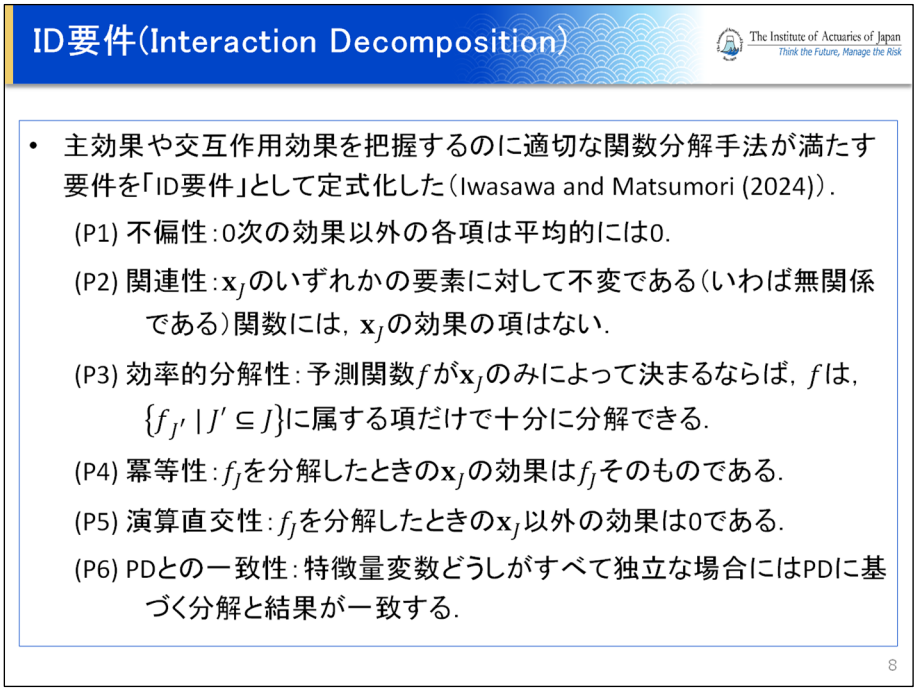
ここで$c_j$は$\mathbb{E}[f_{j,\mathrm{ALE}}(X_j)] = 0$となるように定められる定数．
実用上で$f_{j,\mathrm{ALE}}(x_j)$を推定する式は煩雑なので省略．

- 高次のALEも同様に定義できて，それを基にした関数分解手法も定義できる．
- この関数分解手法も単にALEと呼ぶ．

7

次が、ALE です。Accumulated Local Effects で、PD は後ほどご説明しますが、少し欠点もありますので、ALE という手法も提案されております。これは、1 次元の ALE の式をご紹介いたしますと、今度は定数項なしで、まさに 1 次の部分だけとしてなのですけれども、$f_j(x_j)$として、ややこしいのですが、f を $x_j$ で偏微分し

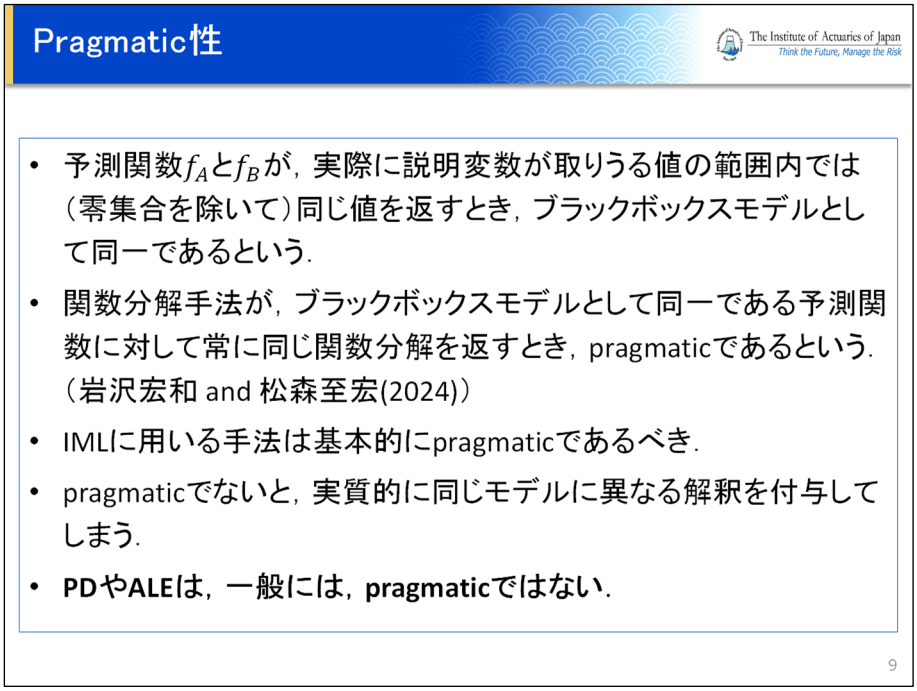
て、1 回期待値を取って、それを積分するというようなものになります。定数がついていますけれども、これは期待値をゼロにするための適当な定数です。偏微分して、その変数ではない方向で期待値を取ったものが平均的な増分として、それを目的のところまで変化分を足していけば、何となく関数が得られて、その関数に沿って影響を与えているというような発想でございます。これは 1 次なのですけれども、これもやはり、高次の ALE も同様に定義できまして、それを基に関数分解することもできます。




あとは、すみません、今、このスライドでは PD と ALE だけ書いたのですけれども、SHAP というものもあります。Shapley Value ですね。local な基本的手法ですけれども、それをたくさん計算して、global な解釈を与えるために使うこともできます。それについては、1 年前の上妻さんのこの年次大会の発表の中で扱われておりますので、そちらもご覧いただければと思います。

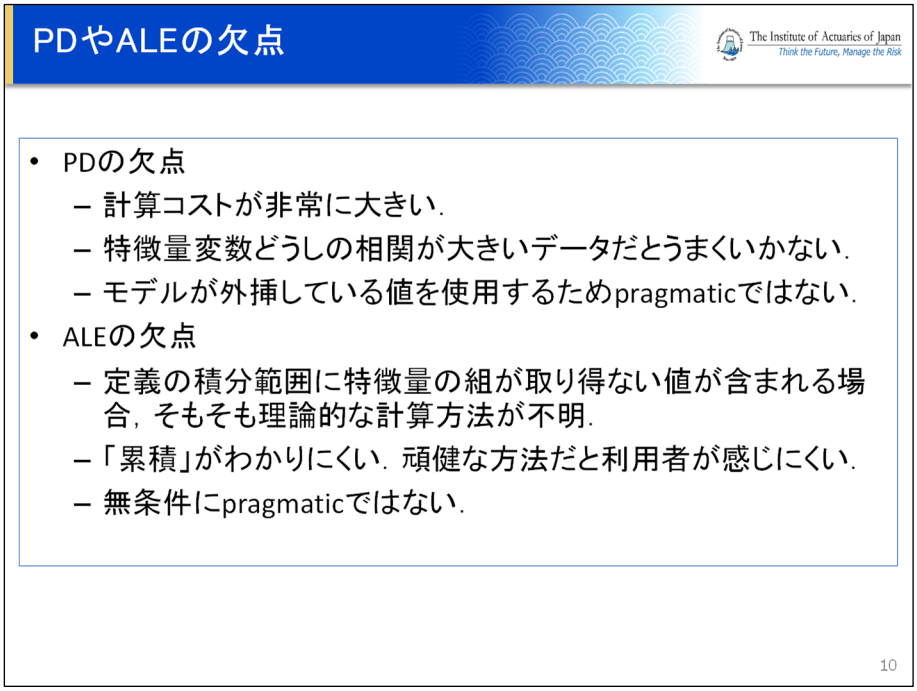
ここから 2 つは、岩沢さんの、私もご協力させていただいているのですけれども、研究の内容をご紹介させていただきます。ALE の方は、関数分解が導入された論文に少し触れられております。ALE は関数分解を与えることができるというものがあるのですけれども、「そもそも、いい関数分解とは何なのか？」というところを岩沢さんが公理化されました。関数分解というものは、何でもいいのですけれども、それが解釈を与えていると思えるためには、どのような条件を満たす必要があるのかというところを、6 つの公理にまとめまして、この 6 つの性質を PD に基づいた関数分解や、ALE に基づいた関数分解が満たすということを証明しております。

一方で、このように公理化すると、もちろん公理を満たすものがたくさんあるのです。公理を満たす関数分解手法を作る方法も、その論文では作るための定理を証明しております。たくさん、実は解釈する方法、解釈手法というものがあるのではないかということです。PD と ALE 以外にももっとたくさんあるのではないかというところを提案といいますか、問題提起しております。




一方で、今の関数分解の公理とは離れた文脈で、もう 1 つ満たしてほしい性質があるのではないかということも考えております。それが、この Pragmatic 性というものです。先ほど、特徴量空間を定義域とする関数であると、ブラックボックスモデルをそう考えると申し上げたのですけれども、機械学習などの文脈では、特徴量の方も確率分布を持っていると考えることが多いと思います。確率分布があるということは、確率分布のサポートがあって、絶対この特徴量の組み合わせは取らない、確率ゼロだというところもあると思います。

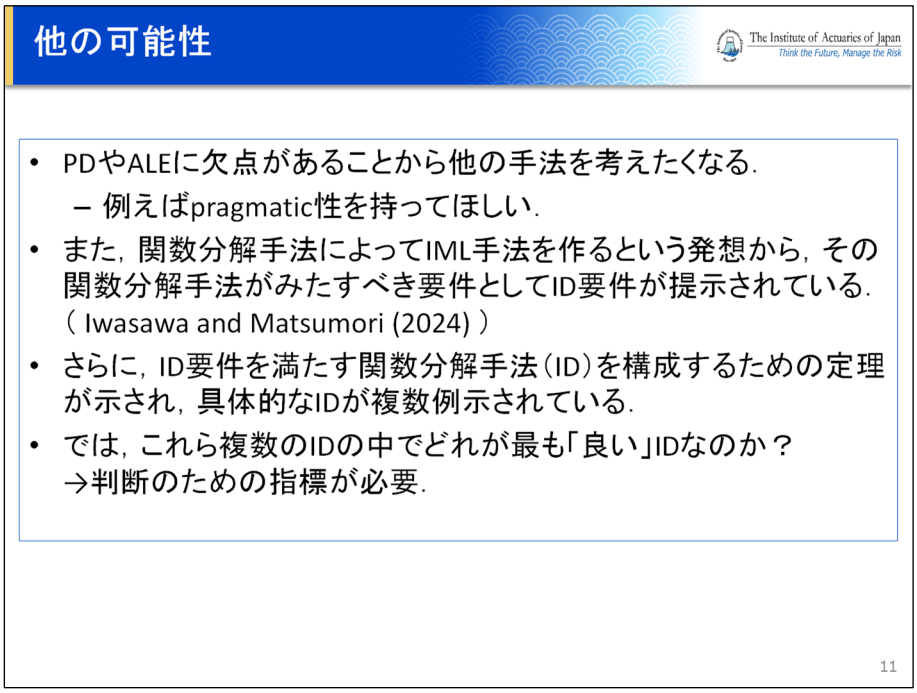
一方で、ブラックボックスモデルはそのようなところのインプットを入れても値が返ってきてしまうのです。ブラックボックスモデルとしては、例えば、正方形の中のどの特徴量の組を入れても値を返してくれるのだけれども、確率分布というものはその中の少し丸い部分しかサポートを持っていないから、特徴量の組はその丸い中しか現実的には起こりませんというときに、丸の中だけで同じだけれども、外で違う関数というものをもちろん作ることができるのですね。そのような関数を作ったときに、その外の部分、確率分布のサポートがないようなゼロの部分のインプット、アウトプットの関係に依存して、違う解釈をしてしまっては困るのではないかと考えております。だから、関数分解をするときに、特徴量の確率分布のサポートの外のふるまいには依らないような、関数分解手法というものを作るべきなのではないかと考えております。先ほどご紹介した、PD や ALE で関数分解を行うと、この Pragmatic 性というものを持たずに、実際には値を取らない外の部分の影響で、分解の結果が変わっていってしまうというようなことがございます。




新しい手法を考えたいということなので、少し先行手法の欠点というところを書いているのですけれども、PD は有名なところで、計算コストが非常に大きいですし、相関があるデータにはうまくいかないというところがあると思います。

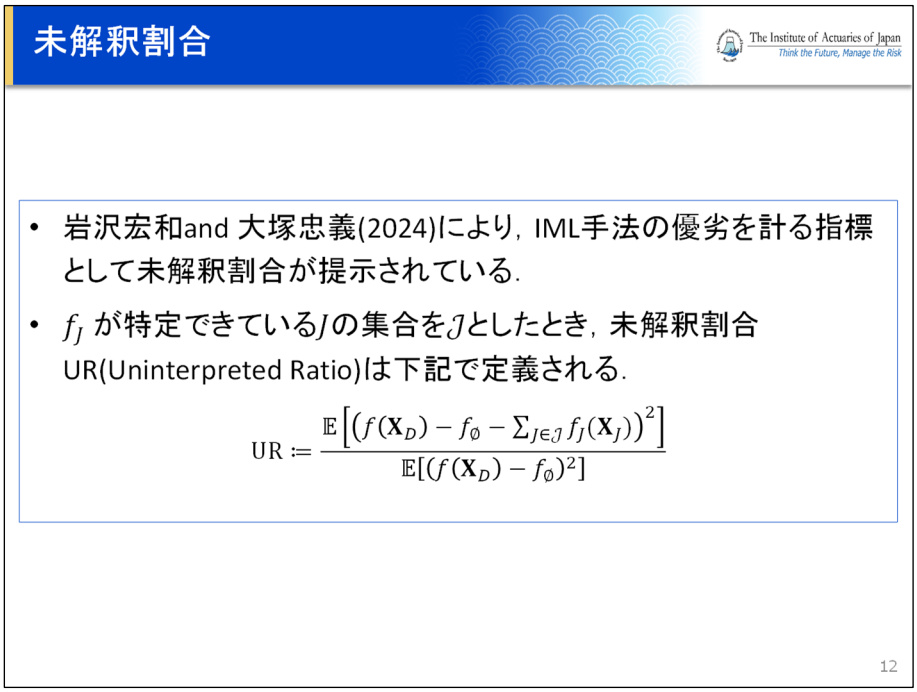
ALE につきましては、私も、明確にこれはうまくいかないという例は持っていないのですけれども、先ほどの定義の偏微分をして、期待値を取って、積分するという方法が、本当にいいのかということがあると思います。これは、ALE を提案した論文で、著者も気をつける必要があると言っております。常に、では、それを使ってモデルを解釈して説明していいのかという不安感があるところかと思っております。

そこで、新しい手法を考えたいとなるのですけれども、もう 1 つ準備をさせていただきます。




では新しい手法を考えたいですと、ID 要件という、先ほどの関数分解の公理を考えました。ID 要件を満た

す分解を作る定理もありましたので、山のように作れるのですけれども、では、どれがよいのですかという指標を考えないといけないと思います。考えないといけないと言うと言いすぎかもしれませんけれども、では、どれを使えばいいのかというものを 1 つ、何か指標を作って評価するということは自然な発想かと思います。




それにつきましては、これまた岩沢先生と大塚先生が、未解釈割合という指標を定義されております。定数部分は、0 次の項は無視して引き算するのですけれども、0 次の項を差し引いた後で、残りの 1 次以上の部分の 2 乗の期待値を指標にしてはどうかという、未解釈割合というものを定義されております。分母は、今、申し上げた、元のモデルから関数分解の定数項を引いて、2 乗の期待値を取ったもので、上は、例えば主効果だけを見たければ、定数項に加えて 1 次の項まで引いて、2 次以上の部分だけにして 2 乗を取って、期待値を取ったものですね。

## MID

The Institute of Actuaries of Japan
Think the Future, Manage the Risk

- 未解釈割合を用いて関数分解手法を比較するのならば，素朴には，未解釈割合を最小化する手法が考えられる．
- $0 \leq k < d$である特定の$k$について，$k$次以下の項のみの合計による平均2乗残差$\mathbb{E}\left[\left(f(\mathbf{X}_D) - \sum_{|J|\leq k} f_J(\mathbf{X}_J)\right)^2\right]$があらゆる関数分解の中で最小であり，各項が中心化されているものを$k$次のMID（Maximal Interpretation Decomposition）と呼ぶ．
- 中心化とは，どの$J' \subset J, |J| \leq k$についても$\mathbb{E}\left[f_J(\mathbf{X}_J)|\mathbf{X}_{J'}\right] = 0$となるようにすること．
- 一意に定まらない場合があるので，$\mathbb{E}\left[\sum_{|J|\leq k} f_J(\mathbf{X}_J)^2\right]$を最小化するするという条件も要請する．
- Pragmaticである．
- **IDではない**．

13

この部分がまだ解釈できていないというように考えて、greedy に取っていくことはどうかと考えることが、すみません、先走ってしまいましたが、MID という手法になります。

## 未解釈割合

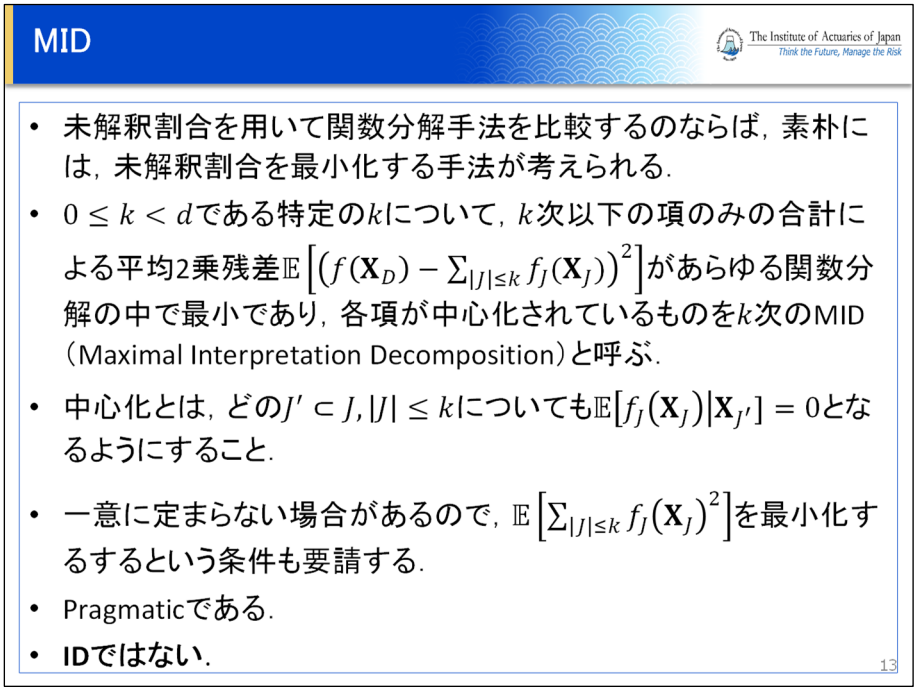
The Institute of Actuaries of Japan
Think the Future, Manage the Risk

- 岩沢宏和and 大塚忠義(2024)により，IML手法の優劣を計る指標として未解釈割合が提示されている．
- $f_J$ が特定できている$J$の集合を$\mathcal{J}$としたとき，未解釈割合UR(Uninterpreted Ratio)は下記で定義される．

$$\mathrm{UR} := \frac{\mathbb{E}\left[\left(f(\mathbf{X}_D) - f_\emptyset - \sum_{J\in\mathcal{J}} f_J(\mathbf{X}_J)\right)^2\right]}{\mathbb{E}[(f(\mathbf{X}_D) - f_\emptyset)^2]}$$

12

今、定義した UR、未解釈割合、Uninterpreted Ratio というものを指標にして、関数分解の優劣を判断したいのであれば、




それを最適化したいという発想で、k 次まで解釈したいのであれば、その k 次までの UR、未解釈割合を最小にするようなものを、シンプルに解釈とすればいいのではないかという発想になります。

これはなかなか一意性が難しいので、まず、中心化ということをして、基本的には期待値がゼロになるような場所を定めて、それでも定まらないようなことがあるので、今度は 2 乗して、全部を足して、括弧の 2 乗の和の期待値を最小にするというような一意性を要求する条件を要請するのですけれども、基本的な発想としては、今、申し上げた、未解釈割合を最小にするという手法になります。これは Pragmatic 性を満たすのですけれども、「紹介したのに何だよ」という話があるのですが、ID 要件は満たさないのです。それは、少し不満であるというところでございます。

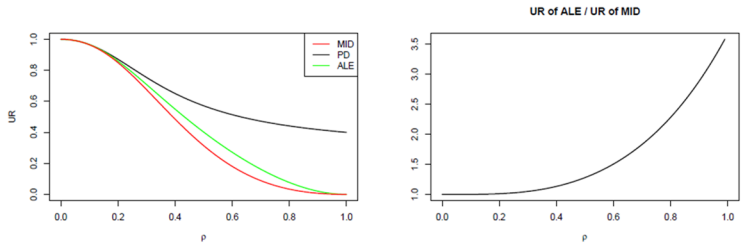
少し細かい話をさせていただきますと、これは存在すれば一意なのですけれども、存在するのかという話はあって、あまり一般的にやると存在しないということもございます。それは、ヒルベルト空間の中で、特徴量空間の L2 空間を考えて、何か 1 点が与えられたときに、その k 次以下の空間が張る空間にプロジェクションすればいいのですけれども、もちろん、相手の空間が Closed Subspace でないと、相手がいないことがあるのです、直交射影の。一般には閉空間ではないのです、この k 次以下の関数の和全体というものが。閉部分空間の和は閉部分空間には必ずしもなりませんので、そのようなものがいないときがあるのですけれども、実用上は問題ないかと個人的には考えております。なぜなら、実際に計算するときは、離散的にやると思いますし、どれだけでも近いものが取れるので、そこの点は大きな問題ではないかと考えております。

## 理論的な例

The Institute of Actuaries of Japan
Think the Future, Manage the Risk

- $X_1, X_2, X_3$が，いずれも平均0，分散1で，全ての共分散が$\rho$の3次元正規分布にしたがっているとする．
- 予測関数$f(x_1, x_2, x_3) = x_1 x_2 x_3$をPD，ALE，MIDそれぞれで1次の項まで関数分解し，$\rho$の値による未解釈割合の挙動を確認する．

$\rho$に応じた未解釈割合の挙動と，MIDの未解釈割合に対するALEの未解釈割合の比率




14

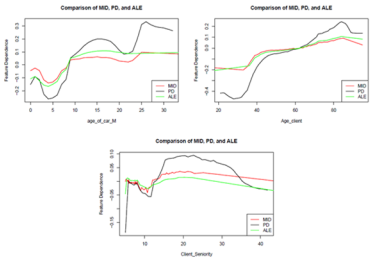
## 実データの例

The Institute of Actuaries of Japan
Think the Future, Manage the Risk

- Insurance data for homeowners and motor insurance customers monitored over five years（スペインの損害保険のデータ）にPD,ALE,MIDを適用．

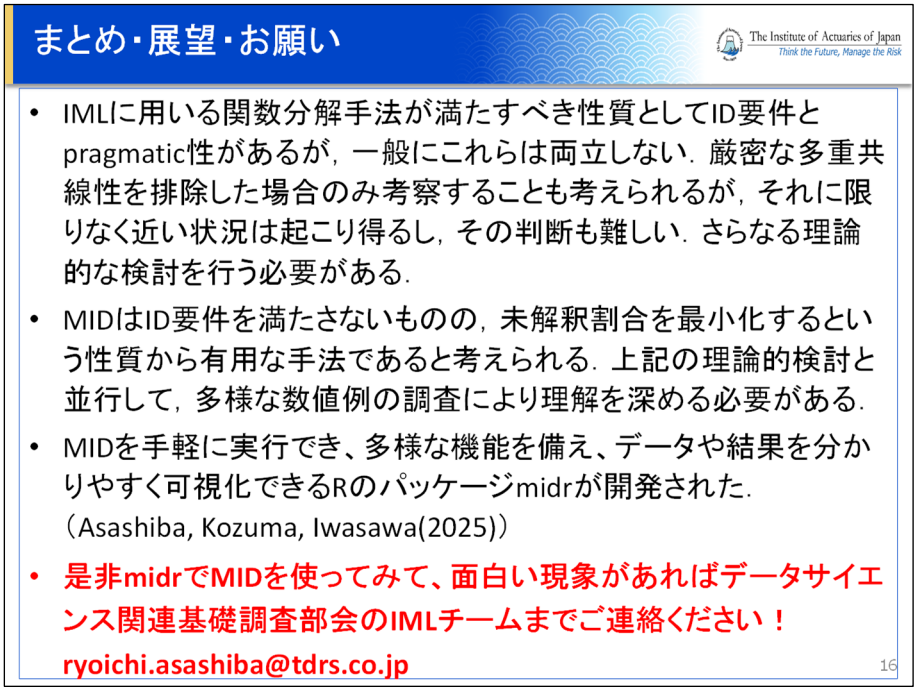
| 手法 | 主効果項までの未解釈割合 | 結果 |
|---|---|---|
| PD | 1超 | 関数分解として機能しない |
| ALE | 0.4732 | 未解釈割合がまだ大きい |
| MID | 0.2235 | 未解釈割合は最小 |

- 重要度の大きい3つの特徴量に関する主効果項の比較
- PDは主効果を過度に捉えている




15

すみません。私の方で、ちょっとした数値例でささやかなものを載せているのですけれども、浅芝さんの方で、後半、きれいなものが出てくると思いますので、それは資料を見ておいてくださいということにさせていただきたいと思います。

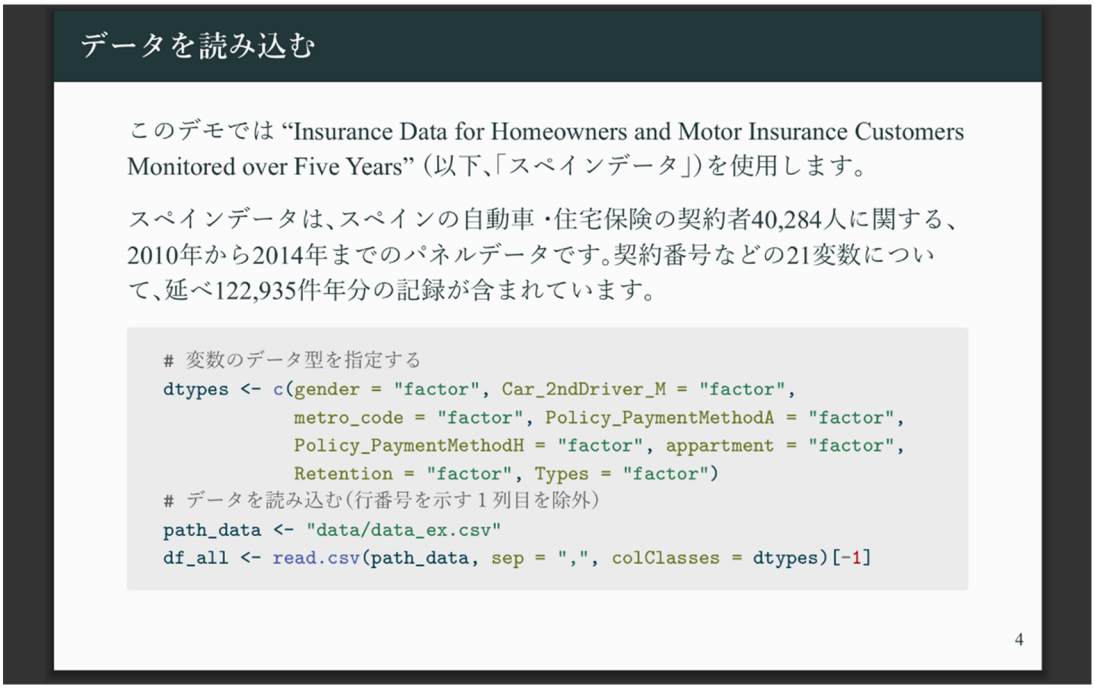



少し押していますが、最後の1枚です。まとめますと、IMLに用いる関数分解手法として、ID要件とpragmatic性というものをご紹介したのですけれども、実はこれは両立しないということが分かっております。どう考えればいいのかということは、まだまだ検討が必要です。

MIDはID要件を満たさないのですけれども、かなり自然であると思います。理論的な検討だけではなくて、数値例の調査が必要かと考えております。そこで出てくるものが、この浅芝さんに今からご紹介していただくパッケージで、非常に多様な機能があって、分かりやすく見えて、手軽に実行できます。皆さん、これをいろいろなモデルに使ってみて、IMLチームまで、浅芝さんのメールアドレスを載せていますので、このようなことがありましたなど、ぜひ、多様な現象を積み上げて調査していきたいと思っておりますので、使っていただいて、どしどしご連絡いただければ幸いでございます。

私の発表は以上です。では、浅芝さんに交代いたします。


**【浅芝】**　ありがとうございます。それでは、ここからは浅芝が発表を引き継がせていただきます。今ご紹介のあった MID を実装した R パッケージである **midr** が、今年の 6 月に R の公式なパッケージ配布サイトである CRAN に無事に登録されました（https://CRAN.R-project.org/package=midr）。そこで、せっかく今年も貴重な機会をいただきましたので、本日はこのパッケージの機能をデモンストレーション形式でご紹介させていただきたいと思います。



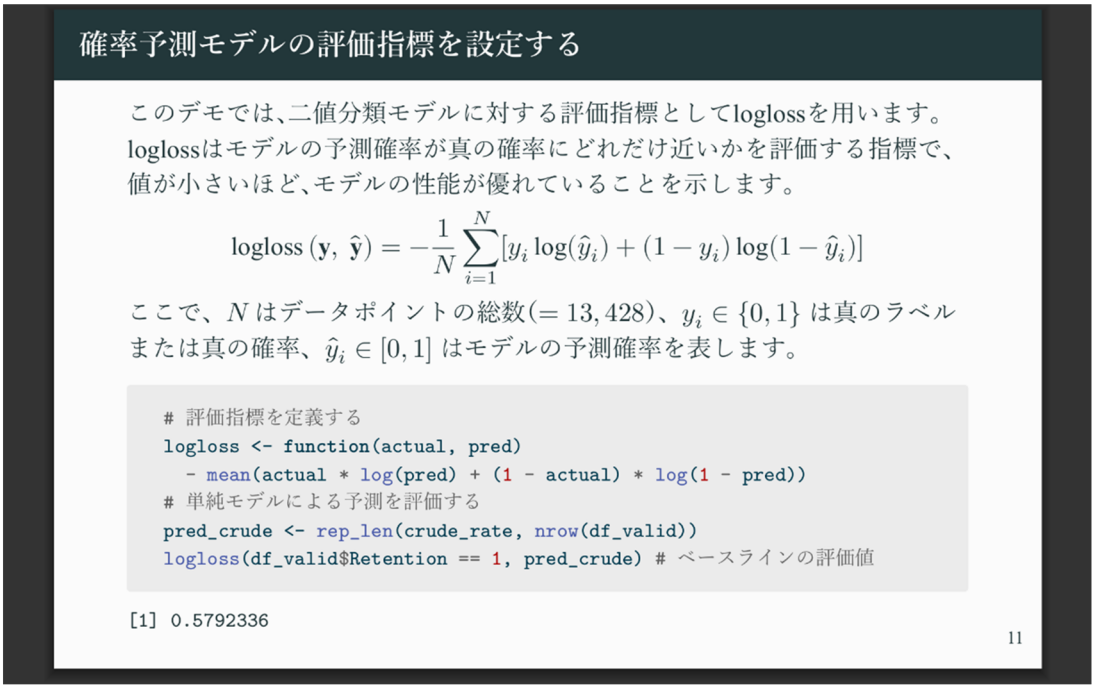

このデモンストレーションでは、スペインの自動車および住宅保険の契約者データを利用します。このデータは、2010 年度から 2014 年度までの計 5 年分のパネルデータで、約 4 万人の契約者に関する、延べ 12 万件年分の記録を含んでいます。変数としては、性別や年齢のような契約者自身の属性の他に、自動車の経過年数や馬力、あるいは各年度において保険金請求があったか、なかったかといった情報が含まれています。

詳細は資料の PDF の方に載っています。


このデータを使って、「各契約が最初の観察年度である 2010 年度に更新されるかどうかに関する確率を予測する」という回帰タスクを設定します。具体的には、観察年度 year が 1 となっている約 4 万件のデータを取り出し、そのうち 2.7 万件の学習用データを用いて契約の更新確率を予測するモデルを構築し、残りの 1.3 万件の検証用データを利用して、作成したモデルの予測精度を評価することにします。


これから予測モデルをいくつか作りますが、それらと比較するためのベースラインモデルとして、ここでは、学習用データに含まれる契約 2.7 万件の全体としての実績更新率を、そのまま、残りの評価用データに対する更新率の予測値として用いる、「粗更新率モデル」のようなものを考えます。学習用データにおける実績更新率を集計すると、およそ 74.3%です。ベースラインモデルでは、この割合 74.3%をそのまま残りの評価

用データに対する予測確率とします。




また、評価用データでの予測精度の評価には、ここでは、一般に「logloss」や「負の対数損失」と呼ばれる指標を利用します。この logloss では、値が小さいほど予測の精度が高いものと考えます。先ほどの平均の更新率に基づくベースラインモデルについて、評価用データにおける実績と予測確率をもとに logloss を計算すると、約 0.579 です。


それでは、モデルを作っていきましょう。まず、予測モデルの例として、GLM の一種であるロジスティック回帰による更新率モデルを構築します。スライドに表示しているコードでモデルを構築でき、構築されたモデルの logloss を評価すると、約 0.5507 となります。これはベースラインモデルで算出した logloss よりも小さくなっているので、このロジスティック回帰モデルは、実績更新率を用いた粗更新率モデルよりも予測

精度の面で少し改善していると言うことができます。

## ロジスティック回帰モデルを解釈する

**midr** パッケージを用いて予測モデルを解釈するには、まず、interpret()関数を用いて対象モデルの「解釈モデル」を構築します。

```
# ロジスティック回帰モデルに関する1次の解釈モデルを構築する
mid_glm <- interpret(
  Retention ~ (. - PolID - year), # モデル式
  data = df_train,                # 学習データ
  model = model_glm,              # 対象モデル
  link = "logit"                  # リンク関数(線形予測子ベースの解釈)
)
```

```
mid_glm$model.class # 対象モデルのクラス
```

```
[1] "glm" "lm"
```

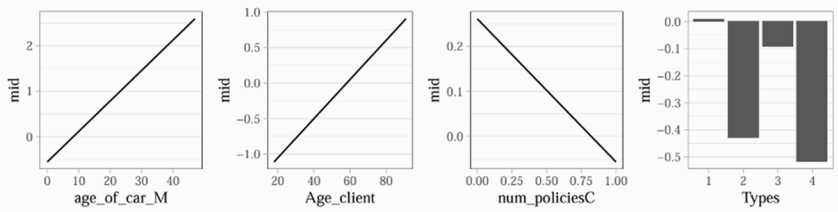
14

ロジスティック回帰モデルは一般に解釈可能なモデルであると考えられています。モデルの中心的要素である線形予測子は、各変数の線形の和にすぎないので、モデルをありのままに理解することもそれほど難しくありません。ですが、ここではあえて **midr** パッケージを利用し、この回帰モデルの解釈を試してみましょう。そのためには、まず、**interpret()** という関数を用いて、対象となる回帰モデル（ロジスティック回帰モデル）の「解釈モデル」を構築します。このとき、リンク関数として logit 関数を指定し、ロジスティック回帰モデルが出力する予測確率そのものではなくて、ロジスティック回帰の中で登場する線形予測子の値を説明する解釈モデルを構築します。**midr** が構築する解釈モデルは加法モデルの一種なので、このようにすると完全な関数分解を達成しやすく、結果の正しさを確認しやすくなります。

## ロジスティック回帰モデルの主効果を確認する（1/3）

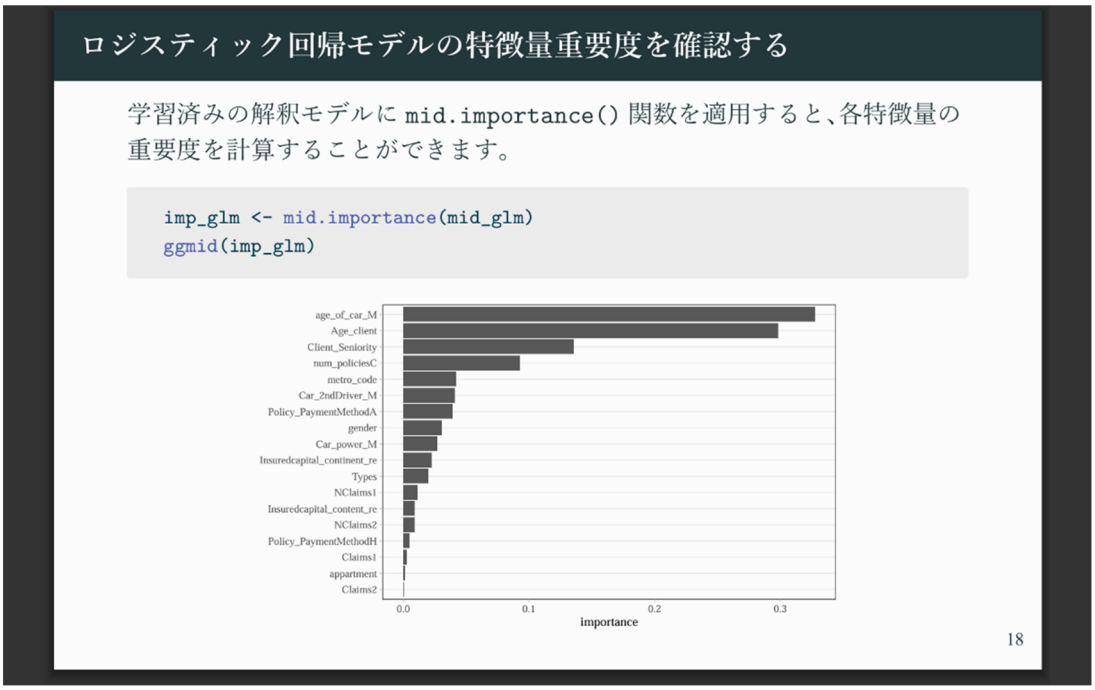
解釈モデルに対して ggmid() 関数や plot() 関数を適用することで、各特徴量の主効果を可視化することができます。

```
grid.arrange(ggmid(mid_glm, "age_of_car_M"),
             ggmid(mid_glm, "Age_client"),
             ggmid(mid_glm, "num_policiesC"),
             ggmid(mid_glm, "Types"),
             nrow = 1)
```




15

**midr** パッケージでは、学習した解釈モデルをもとにして、各変数の主効果や交互作用を簡単に可視化することができます。ロジスティック回帰モデルの線形予測子は、変数の線形の和で表されるので、それぞれの特徴量の効果はすべて線形でなければならないはずです。実際、学習した解釈モデルの主効果を表示してみると、スライドの図のように直線になっていることが分かります。**interpret()** 関数によって、少なくともこの例ではまともな解釈ができているということを、この図から読み取っていただければと思っています。




また、**midr** パッケージでは、解釈モデルに含まれる特徴量の重要度を可視化することも簡単にできるようにしています。なお、その重要度は、各特徴量がモデルの予測値に与える効果の平均的な大きさに基づいて計算をしています。一般に、予測モデルの特徴量の重要度を考えるとき、「評価用データでの損失が、その特徴量を外したときにどれだけ変化するか」というような基準で評価する PFI などの手法が主流であると思います。ここで計算している重要度は、PFI のような重要度指標と異なることがあります。

## ランダムフォレストモデルを構築する

次に、解釈が難しい予測モデルの例として、各契約の更新率を予測するランダムフォレストモデルを構築します。

```
# ランダムフォレスト回帰モデルを構築する
set.seed(42)
model_rf <- ranger(
  Retention ~ (. - PolID - year), data = df_train,
  probability = TRUE, mtry = 5, min.node.size = 250
)
```

```
# 検証データに対する予測確率を得る
preds_rf <- predict(model_rf, df_valid)$prediction[,2L]
# モデルの Log Loss を計算する
logloss(df_valid$Retention == 1, preds_rf) # 0.5353356...
```

```
[1] 0.5353356
```

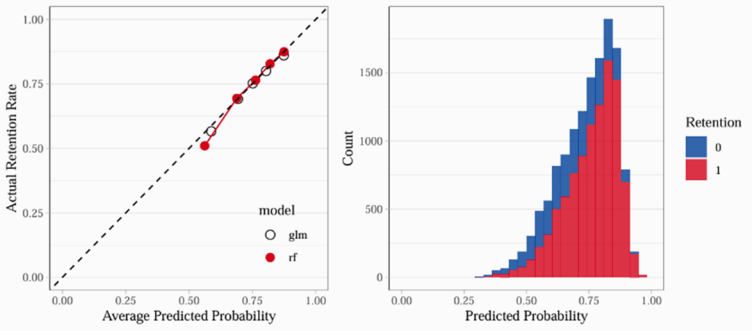
19

次に、ブラックボックスなモデルの例として、ランダムフォレストというアルゴリズムに基づく更新率モデルを構築していきます。ランダムフォレストモデルの構築に **ranger** パッケージを利用しています。スライドに表示しているコードでランダムフォレストモデルを構築し、評価用データに対する logloss を計算すると、約 0.5353 でした。この値は、先ほどのロジスティック回帰モデルの 0.5507 という値よりも低く、したがって、ランダムフォレストモデルの方が logloss 基準で改善していると言うことができます。

## ランダムフォレストモデルを検証する

予測値と正解ラベルをもとに較正プロットとヒストグラムを描きます。

GLM よりメリハリのある予測を行っており、一番予測確率が低かったグループを除けば実績と予測はよく対応しています。




20

ここで、モデルの精度について、少しだけ詳細に確認してみましょう。このスライドで表示しているプロットのうち、左側のプロットはキャリブレーションプロット（較正プロット）と呼ばれるもので、評価用データに対して、GLM（ロジスティック回帰）とランダムフォレストモデルによる予測更新確率と、実際に評価用データで契約が更新された割合（相対頻度）が、どのぐらい正確に対応しているかというものを可視化し

たものです。具体的には、モデルの予測確率に応じて、低い方から高い方に全契約を 5 つのグループに分けて、それぞれのグループに対して、横軸には予測確率の平均値を取り、縦軸には実際の更新率を取ってプロットを作成します。赤い丸と白い丸の系列がありますが、赤い丸がランダムフォレストのもの、白い丸がロジスティック回帰のものです。

比べると、赤い丸のグラフの方がわずかに左側にはみ出ています。これは、ランダムフォレストモデルがロジスティック回帰モデルよりも積極的に低い予測確率を出力していることを示しています。そのグループについては実績更新率が予測確率よりも少し低くなっているようですが、予測確率が多少ずれていても logloss 基準で改善している理由は、ランダムフォレストモデルがこのように少し外れるリスクを取ってでも、低い予測確率を積極的に出力しているためだと考えることができます。

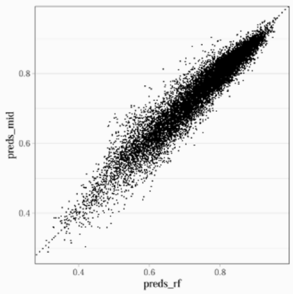

**midr** を使ってランダムフォレストモデルを解釈してみましょう。今度は、ブラックボックスモデルの予測を主効果と 2 次の交互作用に分解する、2 次の解釈モデルを構築します。これは、スライドのコードのように、**interpret()** 関数の最初の引数であるモデル式の部分に 2 乗の指数を加えるだけで実現できます。

## 解釈モデルの性能を評価する

構築した解釈モデルが対象モデルの良い解釈であることを確認するために、
検証データに対する二つのモデルの予測確率の一致度を評価します。

```
# 解釈モデルによる予測確率を得てRMSEを計算する
preds_mid <- predict(mid_rf, df_valid)
sqrt(mean((preds_rf - preds_mid) ^ 2)) # RMSE
```

```
[1] 0.03223142
```

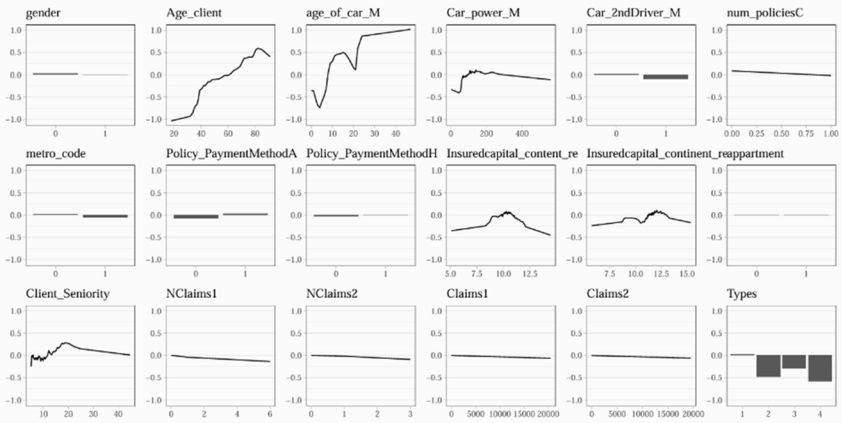



22

構築した解釈モデルはランダムフォレストモデルそのものではなく、予測値を予測する代理モデルに過ぎません。そのため、新しいデータであっても解釈モデルとランダムフォレストモデルが同じような予測をしてくれるかどうか（モデルの**忠実度**）を確認する必要があります。このスライドでは、横軸にランダムフォレストの予測値、y 軸に **midr** による解釈モデルの予測値を取ってプロットしています。少しばらつきはありますが、RMSE で評価すると約 0.032 となります。これが大きいと考えるか小さいと考えるかは微妙なところですが、この解釈モデルは元のブラックボックスモデルの予測をある程度再現できていると考えられます。

## ランダムフォレストモデルの主効果を確認する

すべての変数の主効果を一覧で確認します。




24

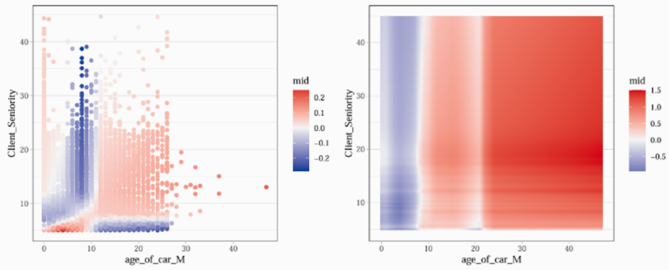
この解釈モデルがランダムフォレストモデルから抽出した各特徴量の主効果を、スライド上に一覧表示しています。例えば、車両の経過年数「age_of_car_M」や契約者年齢「Age_client」のグラフを見ていただくと、モデルが先ほどの線形のグラフではなく、かなり複雑なパターンを学習していることがわかります。例

えば、契約年数が 3 年経過したころにで更新率が非常に大きく低下し、また 10 年経過する頃に更新率が下がる、というようなパターンが捉えられていることが分かります。

## ランダムフォレストモデルの交互作用を確認する

解釈モデルに対して ggmid() 関数や plot() 関数を適用することで、交互作用を可視化することもできます。

```
term_ie <- "age_of_car_M:Client_Seniority"
grid.arrange(nrow = 1,
  ggmid(mid_rf, term_ie, type = "data", theme = "taikai"),
  ggmid(mid_rf, term_ie, main.effects = TRUE, theme = "taikai")
)
```



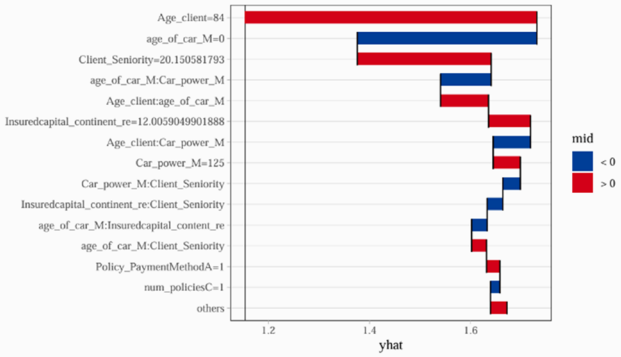

25

また、**midr** パッケージでは 2 変数の交互作用を可視化することもできます。このスライドは、車両の経過年数「age_of_car_M」と、契約継続年数「Client_Seniority」の交互作用を示したものです。左は交互作用のみのプロットで、右は主効果と交互作用を合わせた総効果のプロットです。これらのプロットの右下の方を見てみると、縦軸が表わす契約の継続年数「Client_Seniority」が短い場合、車両の経過年数、横軸の「age_of_car_M」が長くても、1 番下のところに少し青い領域があり、継続年数が短い場合には、車両の経過年数が長くても、更新率が他に、主効果だけの和に比べると、それほど高まらないといったようなことを読み取ることができます。

## ランダムフォレストモデルの予測を分解する

解釈モデルに mid.breakdown() 関数を適用することで、個別の契約に関する予測値を各項の貢献度に分解するプロットを作成できます。

```
mbd <- mid.breakdown(mid_rf, row = 10L)
ggmid(mbd, theme = "taikai@q")
```




27

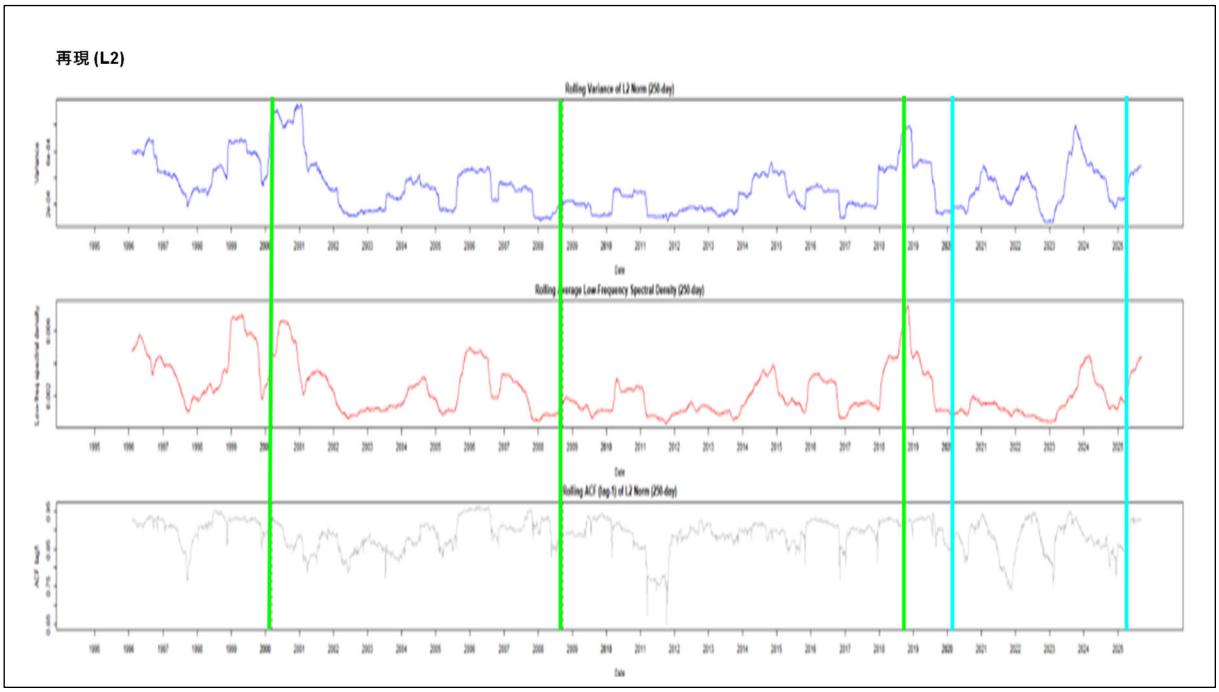
また、**midr** にはいくつかの機能が実装されています。その 1 つとして、breakdown プロットというものをご紹介させてください。これは、ある特定の契約者について、更新確率の予測値がどの変数の影響を受けて構成されているかということを示しています。スタートの黒い縦線が平均の更新確率で、そこから各特徴量の効果で、右に行ったり、左に行ったりしていることを見て取っていただけると思います。具体的には、ここでは、学習用データの中で 10 番目の契約者に対する予測確率に対して、「Age_client」、契約者年齢が 84 歳であることが予測確率を引き上げる一番大きな要因になっていて、一方で、車の経過年数、「age_of_car_M」がゼロ、すなわち新車であることが予測された更新率を引き下げる要因になっているといったことで、個々の予測の根拠を説明することが可能です。


パッケージのデモンストレーションは以上です。最後に簡単にまとめさせていただくと、**midr** パッケージを用いてランダムフォレストのようなブラックボックスとされるモデルの解釈モデルを構築することで、予測精度を維持しながら、そのモデルが学習した非線形な主効果や 2 変数の交互作用を可視化し、解釈することができます。ぜひ、CRAN や GitHub らダウンロードしてお試しいただければ幸いです。また、**midr** は R の基礎的なパッケージとの連携を重視して実装していますが、R で使われている **tidymodels** エコシステムや、Python で使われる **scikit-learn** エコシステムとの連携も一応完成しております。ぜひ、自分のお使いになられているエコシステムの中で MID モデルを構築し、解釈を試みていただければと思います。

私からの発表は以上です。ご清聴ありがとうございました。

**【司会】** それでは、後半の質疑応答に入ります。会場にいる質問のある方は、挙手をお願いします。会場からの質問がないようですので、Slido に投稿されている質問を読ませていただきます。前半部分への質問が 2 点出ていますので、もしよければ、入れ替わった方がいいかもしれない。前半への質問が 2 つ入っています。

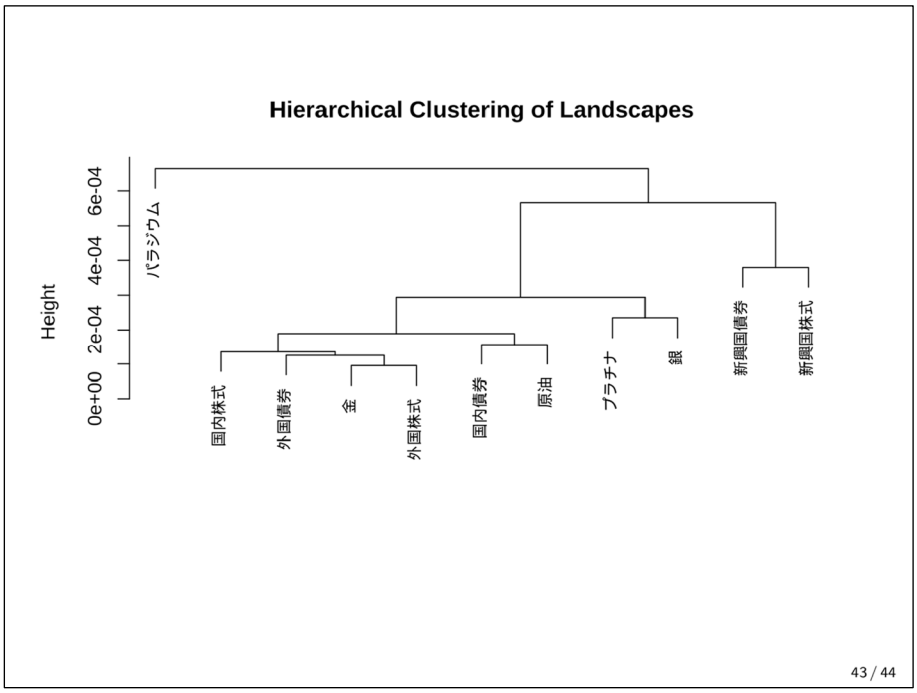
最初は、汪さんへの質問です。「2020 年コロナショックと 2025 年トランプ関税ショックの予兆を検出できなかった理由について何か考察されていますでしょうか。例えば、persistent homology を 2 次まで計算すれば検出精度が向上したりするのでしょうか？」。




**【汪】**　そうですね。2022 年と 2025 年の影響は、主にショックの性質が、前と比べると結構大きい原因ではないかと推測しています。こちらは、特に 2025 年のトランプ関税の政策の原因は人為的なものだと考えており、それは市場の変化では捉えない変化ではないかと思う一方、2020 年のコロナショックも、市場で捉えない原因ではないか。突然発生した、重大な危機の原因によります。以上でいかがでしょう。

**【司会】**　ありがとうございます。

次は、小田さんへの質問になります。




「スライド最後のクラスタリングの結果は結局どのように解釈すればよろしいのでしょうか。クラスタリ

ングが近しいからといって順相関とは限らない（局所的逆相関の場合もクラスタリングが近くなる）と理解したのですが、対象データが局所的逆相関をもつか否かは別途把握しておく必要があるということでしょうか？」。

**【小田】**　そうですね。使い方次第だと思っていて、似たような値動きをしているものに対してクラスタリングをするということは、誰でも思いつくことだし、自然なクラスタリング手法だと思うのですけれども、Persistence Landscape を使った場合は、似たような原因で動く時系列を、近くの似たような要素と見てクラスタリングしていくので、少しクラスタリングの仕方が変わっているということです。それが、実応用上どのようなことに役に立つのかと言われると、それは分析者に委ねられているのかと思っている。でも、これを見ると、例えば新興国の債券と新興国の株式にまとめられていて、新興国の動きだけを切り出されているように見えて、何かの役に立つことも分野によってはあるのかと。私も具体的に何か応用例を思いついているわけではないです。

**【司会】**　ありがとうございます。質問は以上になります。以上をもちまして、セッション B-5「アクチュアリー業務における新たな分析・解釈手法の活用～位相的データ解析および最大解釈分解～」を終了します。発表者にもう一度、盛大な拍手をお願いします。