

回帰型ニューラルネットワークの 圏論的分析

David Sprunger, 勝股 審也

University of Birmingham, 国立情報学研究所

2021/11/10

略歴

専門

- 理論計算機科学
→ プログラミング言語とシステムの意味論

略歴

- エジンバラ大学工学部 PhD (2006)
- 数理解析研究所 助手/助教 (2004-2016)
- 国立情報学研究所 特任准教授 (2017-)
ERATO 蓮尾メタ数理プロジェクト 研究総括補佐
同 メタ理論的統合グループ リーダー

ERATO プロジェクトの理論的基礎

G0 メタ理論的統合グループ

G1 ヘテロジニアス形式手法グループ

G2 産業応用グループ (Waterloo 大学)

G3 インテリジェンス協働形式手法グループ

論理学 様相論理・時相論理・微分動的論理・ホア論理・ロバストネス意味論・超準解析・モデルチェック

確率論 確率的プログラミング言語・グラフィカルモデルの基礎・確率的プログラムの推論・テストケースの効率的な生成・マルチンゲールによる到達可能性解析

圏論 状態遷移系の圏論的モデル・上記論理およびプログラミング言語の意味論・束論

Part I

計算機科学からの圏論

圏論

空間・構造とそれらの間の写像のための抽象代数学

数学の分野	空間・構造	写像	圏
集合論	集合	写像	Set
線形代数	ベクトル空間	線形写像	Vect
位相空間論	位相空間	連続写像	Top
確率論	可測空間	可測写像	Meas
群論	群	準同型	Grp
代数論理	ブール代数	準同型	BA
圏論	↓ 対象	↓ 射	

圏 $\mathbb{C}$ は以下のデータからなる

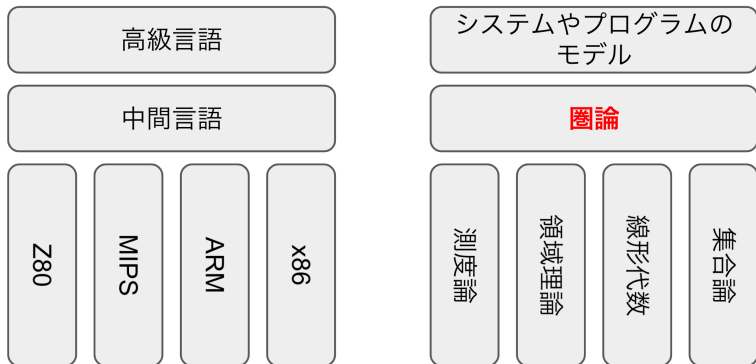
- ① 対象のクラス $|\mathbb{C}|$
- ② 射の集合族 $\{\mathbb{C}(A, B)\}_{A, B \in |\mathbb{C}|}$
- ③ 恒等射 $\text{id}_A \in \mathbb{C}(A, A)$
- ④ 合成 $(\circ) : \mathbb{C}(B, C) \times \mathbb{C}(A, B) \rightarrow \mathbb{C}(A, C)$
 $\text{id}_B \circ f = f \circ \text{id}_A = f, \quad f \circ (g \circ h) = (f \circ g) \circ h$

圏 **Vect**

$$|\mathbf{Vect}| \triangleq \{V \mid V : \text{ベクトル空間}\}$$

$$\mathbf{Vect}(A, B) \triangleq \{f \mid f : A \text{ から } B \text{ への線形写像}\}$$

システム・プログラムの圏論的モデル



- モデルの本質に集中できる
- 数学の差し替え ⇒ モデルや結果の移植

ソフトウェアコンピューティングのモデル

確率的プログラムの圏論的モデル

確率をモデルするモナド Prob を用い射として表現

$$\text{Prog} : \text{Input} \rightarrow \text{Prob}(\text{Output})$$

- プログラミング言語の機構 (ループ、高階関数) と確率論の共存 [Vákár+ POPL '19]
- 確率的プログラムの検証フレームワーク [Barthe+ POPL '12]

ニューラルネットワークの圏論的モデル

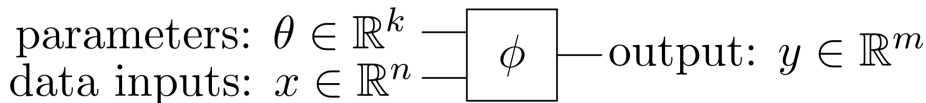
(次のパート)

Part II

Differentiable Causal Computation via Delayed Trace

[David Sprunger, K, LICS'19]

ニューラルネットワーク

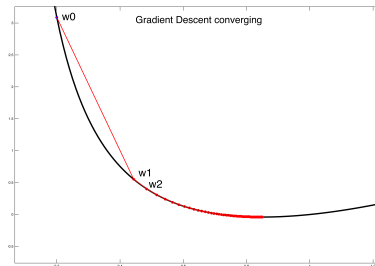


学習: 以下の関数 $e : \mathbb{R}^k \rightarrow \mathbb{R}$ を最小にする θ を見つける

$$e(\theta) \triangleq \sum_{(i,o) \in L} \text{err}(\phi(\theta, i), o)$$

L は学習データ

勾配降下法



<http://www.cs.cornell.edu/courses/cs4780/2015fa/web/lecturenotes/lecturenote07.html>

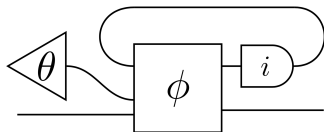
θ を以下で更新する

$$\theta' = \theta - \alpha \times \nabla e(\theta)$$

勾配 ∇e を取るには微分が必要

$$\nabla e(p) = (J e(p))^t(1)$$

Recurrent Neural Network

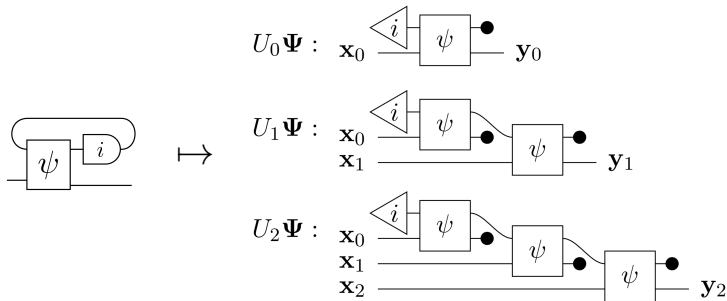


状態関数と出力関数がNNで記述された Mealy 機械

- 入力は時系列データ $x_0, x_1, x_2, \dots \in \mathbb{R}^i$
- 出力は時系列データ $y_0, y_1, y_2, \dots \in \mathbb{R}^o$
- i は初期状態
- $\theta \in \mathbb{R}^k$ はパラメータ

RNN の微分とは何か？

Backpropagation through Time



問い

- 1 展開して微分することの正当性は?
- 2 BPTT はなんらかの意味での微分になっているか?

本研究の貢献

問い

- ① 展開して微分することの正当性は？
- ② BPTT はなんらかの意味での微分になっているか？

我々の貢献

- 遅延フィードバックのある計算のモデルを構成
- 遅延フィードバックのある計算の微分 D^* を定義
- D^* が「展開して微分」に一致

本研究の貢献

問い

- ① 展開して微分することの正当性は？
- ② BPTT はなんらかの意味での微分になっているか？

我々の貢献

- 遅延フィードバックのある計算のモデルを構成: St 構成
- $\mathbb{C}$ が微分圏ならば $St(\mathbb{C})$ も微分圏
- D^* が「展開して微分」に一致

カルテシアン微分圏

...[Blute, Cockett, Seely'09] はカルテシアン圏 $\mathbb{C}$ で

- ① 各対象 X に可換モノイド構造が備わっており

$$1 \xrightarrow{e_X} X \xleftarrow{m_X} X \times X$$

- ② 射の微分 $D : \mathbb{C}(X, Y) \rightarrow \mathbb{C}(X \times X, Y)$ が取れるもの

- ▶ D は第一引数に関して線形
- ▶ 連鎖律が成立
- ▶ 二階微分の性質

Df は直感的には

$$Df(\delta, p) \simeq Jf(p) \cdot \delta$$

カルテシアン微分圏

カルテシアン微分圏 Euc^∞

対象 $0, 1, 2, \dots \in \mathbb{N}$

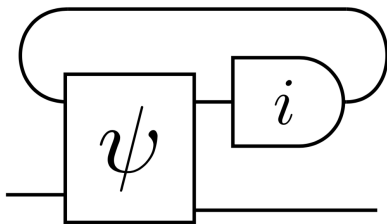
射 $n, m \in \mathbb{N}$ に対し

$$Euc^\infty(n, m) = \{f : \mathbb{R}^n \rightarrow \mathbb{R}^m \mid f_1, \dots, f_m \in C^\infty\}$$

微分 $Df(\delta, p) = Jf(p) \cdot \delta$

射を多項式関数に制限したものも例となる

遅延フィードバック



問い

遅延フィードバックをどう圏論的に表現するか？

我々の提案: St 構成

カルテシアン圏 $\mathbb{C}$ の射に遅延フィードバックを追加する

St 構成

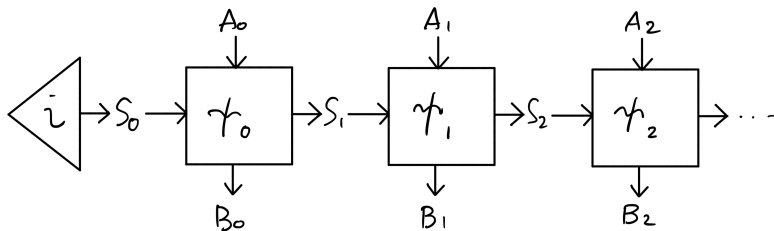
カルテシアン圏 $\mathbb{C}$ に対し、圏 $St(\mathbb{C})$ を以下で定義

対象 $(A_0, A_1, \dots) \in |\mathbb{C}|^\infty$

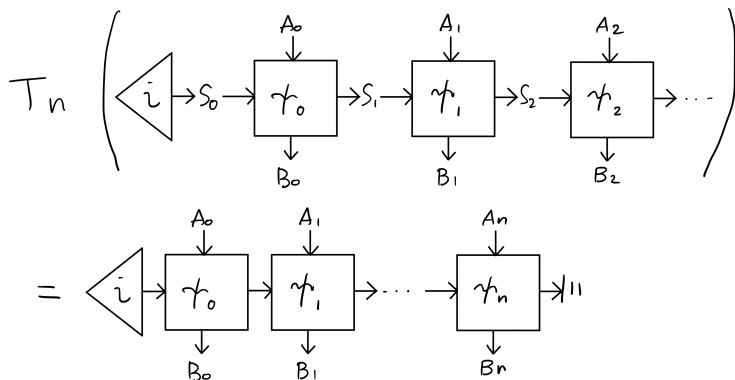
射 $i: 1 \rightarrow S_0$ と

$$\psi_n: S_n \times A_n \rightarrow S_{n+1} \times B_n \quad (n \in \mathbb{N})$$

の組を外延同値性 (次スライド) で割ったもの



射の打ち切り

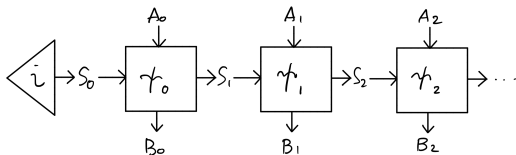


外延同値性

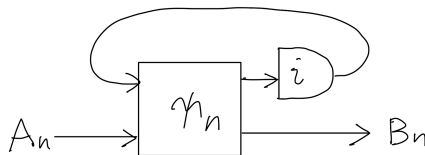
$$(i, \psi) \sim (j, \phi) \stackrel{\text{def}}{\iff} \forall n . T_n(i, \psi) = T_n(j, \phi)$$

射の回路表現

$St(\mathbb{C})$ の射



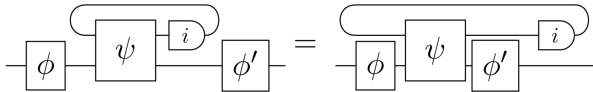
の回路表現を以下で定める。



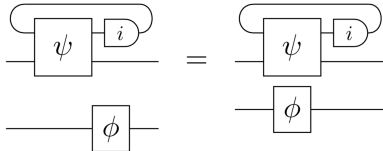
$St(\mathbb{C})$ の射は遅延フィードバックつき回路の時間発展を表現

回路表現どうしの等式

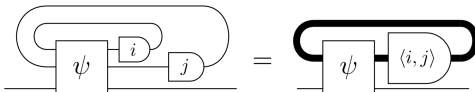
Naturality



Superposing



Vanishing



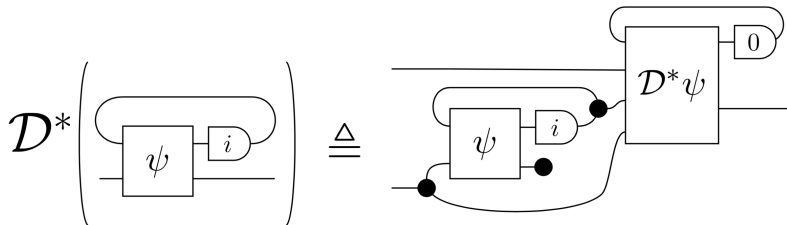
主結果

定理

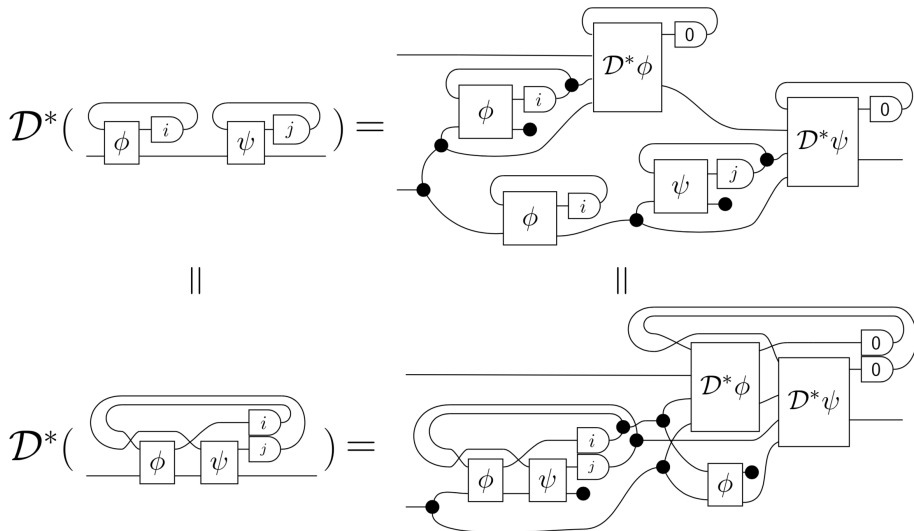
$\mathbb{C}$ がカルテシアン微分圏なら $\text{St}(\mathbb{C})$ に微分演算 D^* が入り、 $\text{St}(\mathbb{C})$ はカルテシアン微分圏となる。

定理 (BPTT is Differential)

$$T_k(D^*(\phi)) = D(T_k(\phi))$$



RNNの微分の連鎖律



まとめ

- RNN の微分操作の圏論的分析
- 遅延フィードバックのモデル: St 構成
- $\mathbb{C}$ がカルテシアン微分圏
 $\implies St(\mathbb{C})$ もカルテシアン微分圏
- $St(\mathbb{C})$ の微分 = 「展開して微分」