Categorical differential structures and their role in abstract machine learning

Geoff Cruttwell
Mount Allison University

---

based on joint work with
(Robin Cockett, Jonathan Gallagher, J.S. Lemay, Benjamin MacAdam, Gordon Plotkin, Dorette Pronk)
and
(Bruno Gavranovic, Neil Ghani, Paul Wilson, Fabio Zanasi)

Topos Institute
July 8th, 2021

Overview

In many supervised/machine learning algorithms, the derivative plays a fundamental role.

- These algorithms usually use gradient descent to get closer to the true value of a function
- If we want to understand what's happening in machine learning abstractly (ie., categorically), it's helpful to have an abstract (categorical) formulation of differentiation
- In this talk I'll begin by discussing one type of categorical differentiation: Cartesian differential categories
- We'll then look at a recent variant of this called Cartesian *reverse* differential categories
- Towards the end of the talk, we'll see why these are useful in developing abstract algorithms that "learn"

One can see this talk as a prelude to Bruno Gavranovic and Paul Wilson's talk at ACT next week!

What is the type of the derivative?

Consider the category **smooth** of Euclidean spaces ($\mathbb{R}^n$'s) and smooth maps between them

- Each map $f : \mathbb{R}^n \longrightarrow \mathbb{R}^m$ in this category has its associated Jacobian, which at a point $x \in \mathbb{R}^n$ gives an $n \times m$ matrix, ie., a linear map from $\mathbb{R}^n \longrightarrow \mathbb{R}^m$
- One can think of this operation as a map

$$J[f] : \mathbb{R}^n \longrightarrow \mathrm{Lin}[\mathbb{R}^n, \mathbb{R}^m]$$

- Alternatively, by uncurrying, we can think of it as a map

$$D[f] : \mathbb{R}^n \times \mathbb{R}^n \longrightarrow \mathbb{R}^m$$

  (the directional derivative)
- For example, if $f : \mathbb{R}^2 \longrightarrow \mathbb{R}$ is defined by $f(x_1, x_2) = x_1^2 x_2 + \sin(x_2)$,

$$D[f](x_1, x_2, x_1', x_2') = (2x_1x_2) \cdot x_1' + (x_1^2 + \cos(x_2)) \cdot x_2'$$

- Thus, in this category, for any map $f : A \longrightarrow B$, we have an associated map

$$D[f] : A \times A \longrightarrow B$$

  satisfying certain properties

Cartesian left additive categories

To talk about some of the properties of differentiation, we will need our base category to have a bit of pre-existing structure:

- A **Cartesian** category is a category with finite products
- A **left additive** category is a category $\mathbb{X}$ in which each homset $\mathbb{X}(A, B)$ has the structure of a commutative monoid, and addition is preserved by post-composition:

$$f; (g + h) = f; g + f; h^{1} \text{ and } f; 0 = 0$$

- A map f in a left additive category is said to be **additive** if it preserves addition:

$$(g + h); f = g; f + h; f \text{ and } 0; f = 0$$

- A **Cartesian left additive category** is a category which is Cartesian and left additive and these structure are compatible, eg., all projections are additive

[1]We are using ; to represent diagrammatic order of composition

Cartesian differential categories

Definition (Blute/Cockett/Seely 2009)

A **Cartesian differential category** consists of a Cartesian left additive category $\mathbb{X}$ in which for every map $f : A \longrightarrow B$ there is an associated map

$$D[f] : A \times A \longrightarrow B$$

satisfying seven axioms:

- **[CD.1]** $D[0] = 0$ and $D[f + g] = D[f] + D[g]$
- **[CD.2]** $\langle x, 0 \rangle; D[f] = 0$ and

$$\langle x, v_1 + v_2 \rangle; D[f] = \langle x, v_1 \rangle; D[f] + \langle x, v_2 \rangle; D[f]$$

- **[CD.3]** $D[1] = \pi_1,\ D[\pi_0] = \pi_1; \pi_0,\ D[\pi_1] = \pi_1; \pi_1$
- **[CD.4]** $D[\langle f, g \rangle] = \langle D[f], D[g] \rangle$

Cartesian differential categories continued

Definition

- **[CD.5]** the chain rule: for composable maps f, g,

$$D[f; g] = \langle \pi_0; f, D[f] \rangle; D[g]$$

- **[CD.6]** linearity of the derivative:

$$\langle x, v, 0, w \rangle D[D[f]] = \langle x, w \rangle D[f]$$

- **[CD.7]** symmetry of mixed partial derivatives:

$$\langle x, v_1, v_2, w \rangle D[D[f]] = \langle x, v_2, v_2, w \rangle D[D[f]]$$

Examples

Examples of Cartesian differential categories (CDCs):

- **smooth**
- Polynomial functions between $\mathbb{R}^k$'s
- $\mathbb{Z}_n$ polynomials between finite $\mathbb{Z}_n^k$'s
- Convenient vector spaces (a form of infinite-dimensional calculus)
- Abelian functor calculus[2]

CDCs are related to many other categorical theories of differentiation:

- The Euclidean R-modules in a model of synthetic differential geometry form a CDC
- A model of the differential λ-calculus is a CDC
- The coKleisli category of a differential category is a CDC
- A Fermat theory is a CDC

[2] See "Directional derivatives and higher order chain rules for abelian functor calculus" by BJORT

Linearity in a CDC

Definition

A map $f : A \longrightarrow B$ in a CDC is said to be **linear** if

$$D[f] = \pi_1; f.$$

Eg., in **smooth**, this agrees with the ordinary (vector space) notion of linear.

Definition

A map $f : A \times B \longrightarrow C$ in a CDC is **linear in its second variable** if

$$\langle \pi_0, \pi_1, 0, \pi_2 \rangle; D[f] = \langle \pi_0, \pi_2 \rangle; f$$

[CD.6] is equivalent to asking that for any $f : A \longrightarrow B$,

$$D[f] : A \times A \longrightarrow B$$

is linear in its second variable.

The simple fibration and CDCs

Definition

If $\mathbb{X}$ is a Cartesian category, the **simple fibration over** $\mathbb{X}$, written as $\mathbb{X}[\mathbb{X}]$, is the category with objects pairs (A, A') and maps pairs $(f, f') : (A, A') \longrightarrow (B, B')$ where

$$f : A \longrightarrow B \text{ and } f' : A \times A' \longrightarrow B'.$$

The composite of (f, f') with (g, g') is given by $f; g$ with

$$\langle \pi_0; f, f' \rangle; g'.$$

Definition

If $\mathbb{X}$ is a Cartesian differential category, $\mathrm{Lin}[\mathbb{X}]$ is the subcategory of the simple fibration consisting of maps $(f, f') : (A, A') \longrightarrow (B, B')$ such that $f' : A \times A' \longrightarrow B'$ is linear in its second variable.

There are forgetful functors $U : \mathbb{X}[\mathbb{X}] \longrightarrow \mathbb{X}$ and $U_L : \mathrm{Lin}[\mathbb{X}] \longrightarrow \mathbb{X}$ which are both fibrations.

The simple fibration and CDCs continued

Lemma

If $\mathbb{X}$ is a CDC, $\mathbb{X}$ has a section D of the fibration $U_L : Lin[\mathbb{X}] \longrightarrow \mathbb{X}$ given by sending

$$A \mapsto (A, A)$$

and

$$(f : A \longrightarrow B) \mapsto (f, D[f]) : (A, A) \longrightarrow (B, B).$$

In fact, functoriality of this section is precisely the chain rule!

Tangent categories

Note that CDCs are not sufficient for differential geometry: for example, the category of smooth manifolds is not a CDC.

- In the category of smooth manifolds, every object M has an associated "tangent bundle" TM
- This operation is functorial: given any map $f : M \to N$, there is an associated map
$$T(f) : TM \to TN$$
which is the analogue of the derivative between Euclidean spaces
- This structure is abstracted by **tangent categories** which involve asking for a category $\mathbb{X}$ with an endofunctor $T : \mathbb{X} \to \mathbb{X}$ equipped with various natural transformations which the tangent bundle on smooth manifolds possesses
- CDCs are essentially tangent categories in which every tangent bundle is trivial, ie., for each A
$$T(A) \cong A \times A,$$
one recovers $D[f]$ from this as the composite
$$A \times A \cong T(A) \xrightarrow{T(f)} T(B) \cong B \times B \xrightarrow{\pi_1} B$$

Reverse differentiation

This is all good...but most machine learning algorithms use the so-called "reverse" mode of differentiation!

- Recall that the Jacobian of $f : \mathbb{R}^n \to \mathbb{R}^m$ at a point of $\mathbb{R}^n$ gives a linear map from $\mathbb{R}^n$ to $\mathbb{R}^m$, and we get a map

$$D[f] : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^m.$$

- *Reverse* differentiation uses the *transpose* of the Jacobian, which is a linear map $\mathbb{R}^m \to \mathbb{R}^n$, and this gives a map

$$R[f] : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n.$$

- Note the difference in type from $D[f]$!
- If $f : \mathbb{R}^2 \to \mathbb{R}$ is defined by $f(x_1, x_2) = x_1^2 x_2 + \sin(x_2)$,

$$D[f](x_1, x_2, x_1', x_2') = (2x_1x_2) \cdot x_1' + (x_1^2 + \cos(x_2)) \cdot x_2'$$

while

$$R[f](x_1, x_2, y') = [(2x_1x_2) \cdot y', (x_1^2 + \cos(x_2)) \cdot y']$$

CDCs vs. RDCs

- Thus,from a map $f : A \to B$, the "forward" derivative is a map of type
$$D[f] : A \times A \to B$$
while the reverse derivative is a map of type
$$R[f] : A \times B \to A.$$
- There is no reason why a CDC should have a reverse derivative.
- We could get one in a similar way to how the reverse derivative for **smooth** is defined: ask for a "dagger structure on linear maps" and define $R[f]$ as the dagger of $D[f]$ (in its second variable)
- An alternative is to axiomatize the resulting structure on its own, leading to **Cartesian reverse differential categories**
- We'll look at each of these possibilities in turn

The dual of the simple fibration

Any fibration has an associated *dual* fibration given by taking the opposite category of each fibre.

Definition

If $\mathbb{X}$ is a Cartesian category, the **dual of the simple fibration** (also known as the category of **lenses**!) is the category whose objects are pairs (A, A') and maps $(f, f^*) : (A, A') \longrightarrow (B, B')$ consist of a pair of maps

$$f : A \longrightarrow B, f^* : A \times B' \longrightarrow A'$$

with the composite of (f, f^*) with (g, g^*) given by $f; g$ with

$$\langle \pi_0, \langle \pi_0; f, \pi_1 \rangle; g^* \rangle f^*.$$

The dual of the linear fibration, $\mathrm{Lin}^*[\mathbb{X}]$ is the same as above, but requires that f^* be linear in its second variable.

Contextual linear dagger

Definition

If $\mathbb{X}$ is a CDC, say it has a **contextual linear dagger** if there is an identity-on-objects fibration functor

$$()^\dagger : \mathsf{Lin}[\mathbb{X}] \longrightarrow \mathsf{Lin}^*[\mathbb{X}]$$

which when composed "with itself" gives the identity.

For example, **smooth** has such structure given by taking the transpose. The effect of having a contextual linear dagger is that given a map

$$f : A \times A' \longrightarrow B'$$

which is linear in its second variable, one gets

$$f^\dagger : A \times B' \longrightarrow A'$$

also linear in its second variable, and $(f^\dagger)^\dagger = f$.

Contextual linear dagger ctd.

If $\mathbb{X}$ is a CDC with a contextual linear dagger, then every map

$$f : A \longrightarrow B$$

has an associated map

$$D[f] : A \times A \longrightarrow B$$

which is linear in its second variable, and so also has a map

$$D[f]^\dagger =: R[f] : A \times B \longrightarrow A$$

which is linear in its second variable, and can be thought of as the "reverse derivative" of f.

Question: what properties does this operation satisfy?

CRDC definition

Definition (Cockett et. al. 2020)

A **Cartesian reverse differential category** (CRDC) consists of a left additive category $\mathbb{X}$, which has, for every map $f : A \longrightarrow B$, a map

$$R[f] : A \times B \longrightarrow A$$

satisfying seven axioms:

- **[RD.1]** $R[0] = 0$ and $R[f + g] = R[f] + R[g]$
- **[RD.2]** $\langle x, 0\rangle; R[f] = 0$ and

$$\langle x, v_1 + v_2\rangle; R[f] = \langle x, v_1\rangle; R[f] + \langle x, v_2\rangle; R[f]$$

- **[RD.3]** $R[1] = \pi_1$, $R[\pi_0] = \pi_1\iota_0$, $R[\pi_1] = \pi_1\iota_1$ where $\iota_0 = \langle 1, 0\rangle$ and $\iota_1 = \langle 0, 1\rangle$
- **[RD.4]**

$$R[\langle f, g\rangle] = (1 \times \pi_0); R[f] + (1 \times \pi_1)R[g]$$

CRDC definition ctd.

Definition

- **[RD.5]** reverse chain rule:

$$R[f;g] = \langle \pi_0, \langle \pi_0; f, \pi_1 \rangle; R[g] \rangle; R[f]$$

- **[RD.6]** linearity of the derivative:

$$(1 \times \pi_0, 0 \times \pi_1); (\iota_0 \times 1); R[R[R[f]]]\pi_1 = (1 \times \pi_1); R[f]$$

- **[RD.7]** symmetry of mixed partials:

$$(\iota_0 \times 1;)R[R[\iota_0 \times 1); R[R[f]]; \pi_1]]; \pi_1 =$$

$$\mathrm{ex}; (\iota_0 \times 1); R[R[(\iota_0 \times 1); R[R[f]]; \pi_1]]; \pi_1$$

where ex exchanges the middle two terms.

Just as a CDC gives a section of the simple fibration, so a CRDC gives a section of the dual of the simple fibration (ie., the category of lenses).

Examples

Any CDC with a contextual linear dagger is a CRDC. Examples:

- **smooth**
- Polynomial functions between $\mathbb{R}^n$'s
- $\mathbb{Z}_n$ polynomials between $\mathbb{Z}_n^k$'s

We're working on adding more examples.

The CDC hidden inside a CRDC

To go from a CDC to a CRDC, one needs dagger structure. But one doesn't need any additional structure to go from a CRDC to a CDC!

- Suppose $\mathbb{X}$ is a CRDC, and let $f : A \longrightarrow B$, so that

$$R[f] : A \times B \longrightarrow A.$$

- Then

$$R[R[f]] : A \times B \times A \longrightarrow A \times B$$

- And we can extract a forward derivative from this by inserting 0's and projecting: define $D[f] : A \times A \longrightarrow B$ by

$$D[f] = \langle \pi_0, 0, \pi_1 \rangle; R[R[f]; \pi_0.$$

- This satisfies all the rules to have a CDC!

(This "trick" is somewhat well-known in the automatic differentiation community.)

Characterization of CRDCs

In a similar way one can show that the resulting CDC has a contextual linear dagger (again by using the reverse derivative to define the dagger). Then we get

Theorem (Cockett et. al. 2020)

The following are equivalent:

- *A CDC with a contextual linear dagger*
- *A CRDC*

Basics of supervised learning

- In supervised learning, one wants to learn some objective function

$$o : A \longrightarrow B$$

- To do this, one fixes a parameter space P and builds a function

$$f : P \times A \longrightarrow B$$

  (the "neural network")
- One hopes that for some value of p, $f(p, -) : A \longrightarrow B$ will closely approximate o.
- One starts with some value p_0, and then performs some iterative process to get new values $p_1, p_2, \ldots$.
- The iterative process often involves some training data: values of the function that one knows $b_i = o(a_i)$

The reverse derivative and supervised learning

The reverse derivative of the network f is a key component in gradient-based learning algorithms.

- If one has $f : P \times A \longrightarrow B$, then

$$R[f] : P \times A \times B \longrightarrow P \times A$$

- Note that this seems like exactly the right type to do learning!
- One can feed into this function the current parameter p and the current training data pair (a_i, b_i) and get back a new value of P (and a value of A, which is related to so-called "deep dreaming")
- In fact, it's a little bit more complicated than that, as the $R[f]$ expects to see a *change* in B and gives back a *change* in P...
- This is where gradient descent and the loss function come into play
- But because of its bidirectional type, the reverse derivative of f plays a key role in these algorithms

CRDCs and supervised learning

Bruno Gavranovic and Paul Wilson will talk at ACT next week about more of these details, showing how to talk about gradient-based supervised learning algorithms in any CRDC. The framework is quite general:

- It allows for different types of gradient descent algorithms such as momentum and Adagrad
- It allows for different types of loss functions
- It allows one to change the base category to any CRDC, encompassing learning on Boolean circuits developed by Wilson and Zanasi

Conclusions

- CDCs and CRDCs generalize different types of differentiation operations between Euclidean spaces
- They have some interesting theoretical aspects: for example, a CRDC "contains a CDC inside of it" (but not the converse)
- CRDCs turn ordinary maps into lenses, which can "learn"
- CRDCs can be used to talk about gradient-based learning algorithms in different settings
- Future work: developing the tangent category analogue of reverse differentiation, ie., "cotangent categories". Will hopefully be useful in understanding learning on manifolds.

References

- (2019) B. Fong, D. Spivak, and R. Tuyeras. **Backprop as functor: a compositional perspective on supervised learning**. In *Proceedings of LICS 2019*.
- (2020) R. Cockett, G. Cruttwell, J. Gallagher, J.S. Lemay, B. MacAdam, G. Plotkin, and D. Pronk. **Reverse derivative categories**. In *Proceedings of CSL 2020*.
- (2020) P. Wilson and F. Zanasi. **Reverse derivative ascent: a categorical approach to learning boolean circuits**. In *Proceedings of ACT 2020*.
- (2021) G. Cruttwell, B. Gavranonvic, N. Ghani, P. Wilson, and F. Zanasi. **A categorical framework for gradient-based learning**, arXiv:2103.01931.