HIGHER ORDER DIFFERENTIAL CALCULUS IN MATHLIB

SÉBASTIEN GOUËZEL

Abstract. We report on the higher-order differential calculus library developed inside the Lean mathematical library `mathlib`. To support a broad range of applications, we depart in several ways from standard textbook definitions: we allow arbitrary fields of scalars, we work with functions defined on domains rather than full spaces, and we integrate analytic functions in the broader scale of smooth functions. These generalizations introduce significant challenges, which we address from both the mathematical and the formalization perspectives.

1. Introduction

Calculus is a basic and central topic in mathematics, often first studied for functions from $\mathbb{R}$ to $\mathbb{R}$, then from $\mathbb{R}^n$ to $\mathbb{R}^m$, then for functions between normed vector spaces – where the added abstraction and generality are needed for many important applications. The basic idea of calculus is to approximate locally a function by a linear map, its differential. Often, this is not enough, and one should approximate a function by a polynomial or, equivalently, study iterated differentials, i.e., the differential of the differential, and so on. This is higher order calculus.

In this article, we describe how higher order calculus has been formalized in the `mathlib` library [mat20] for the Lean proof assistant [dMKA$^+$15], insisting on the motivation for several design choices that differ from mainstream definitions in classical textbooks. In a nutshell, these choices stem from the broad range of applications that our formalization should make possible, from Lie groups over p-adic fields to Schwartz spaces for distribution theory.

First-order differential calculus is the study of derivatives of maps. In this respect, the main textbook definition is the following:

Definition 1.1. *Let E and F be two real normed spaces. A map $f : E \to F$ is differentiable at a point $x \in E$ if there exists a continuous linear map L from E to F such that, as y tends to x,*

$$f(y) = f(x) + L(y - x) + o(y - x).$$

The $o()$ notation means here that $\|f(y) - f(x) - L(y - x)\|/\|y - x\|$ tends to zero as y tends to x while being different from x.

When f is differentiable at a point x, the linear map L from the definition is unique. It is the (Fréchet) *differential* (or *derivative*) of f at x, and is often denoted by $Df(x)$ or $D_x f$ or $f'(x)$.

Date: September 4, 2025.

Key words and phrases. Formalization, Lean, Mathlib, differential calculus, smoothness classes, analytic functions.

First-order differential calculus has been formalized by Avigad in `mathlib` essentially by following the Isabelle/HOL implementation described in [HIH13, Section 5.2]. While this is not the main focus of this article, we will recap in Section 2 some features of this implementation. See also [ACR18, Section 4.5] for a formalization of derivatives in the MathComp Analysis library for Rocq.

For notions of higher-order differentiability, we will follow for instance the reference textbook [Car71, Section 5.3]. Let us denote by $\mathcal{L}(E, F)$ the space of continuous linear maps from E to F, when E and F are real normed vector spaces. With the operator norm, $\mathcal{L}(E, F)$ is also a real normed vector space. Informally, a function has n degrees of smoothness if it has n successive derivatives which are all continuous. We will say in this case that the function is C^n. More formally, the standard definition of higher-order differentiability is the following inductive definition.

Definition 1.2. *Let E and F be two real normed spaces, and let $f : E \to F$. We say that f is C^0 if it is continuous. For a natural number $n > 0$, we say that f is C^n if it is differentiable and its derivative $Df : E \to \mathcal{L}(E, F)$ is C^{n-1}. We say that f is C^∞ if it is C^n for all $n \in \mathbb{N}$.*

When f is C^2, its second derivative $D^2 f(x)$ is well defined, as an element of $\mathcal{L}(E, \mathcal{L}(E, F))$. There is a canonical (uncurrying) identification between $\mathcal{L}(E, \mathcal{L}(E, F))$ and the space of continuous bilinear maps from $E \times E$ to F, so $D^2 f(x)$ is usually rather seen as a continuous bilinear map. In the same way, $\mathcal{L}(E, \mathcal{L}(E, \cdots \mathcal{L}(E, F) \cdots)$ is identified with the space of continuous multilinear maps from E^n to F, so the iterated n-th derivative of f, denoted $D^n f(x)$, is usually seen as a continuous multilinear map.

`mathlib` is an integrated mathematics library, whose ultimate goal is to be able to formalize all current mathematical research. Formalization of notions in `mathlib` should therefore be designed to allow all important applications down the line. A central application of higher order calculus is the theory of manifolds. Among the different kinds of manifolds encountered in the literature, let us mention manifolds with boundary (they are basic blocks in Thurston's program to classify 3-dimensional manifolds, solved in the positive by Perelman and giving Poincaré conjecture in dimension 3), manifolds over the p-adic numbers (especially important to define p-adic Lie groups), Banach manifolds (key to study groups of diffeomorphisms of compact real manifolds). For these applications, Definition 1.2 is not enough. Let us highlight the limitations of this definition.

(1) It is only given over the real numbers. For p-adic Lie groups, one needs higher-order differential calculus over an arbitrary field.
(2) It is only written in the whole space (or, by a straightforward generalization, open sets of the vector space). To study manifolds with boundary, we need a notion of smooth functions in a closed half-space (or in a closed quadrant for manifolds with corners). More generally, we should define C^n functions on subsets of vector spaces.
(3) Calculus over normed fields different from $\mathbb{R}$ or $\mathbb{C}$ is typically only well behaved for analytic functions (see Section 5 for more on this). For this reason, Bourbaki [Bou71] studies manifolds of class C^n for $n \in \mathbb{N} \cup \{\infty, \omega\}$ (where C^ω stands for analytic functions), and assumes when the field is not $\mathbb{R}$ or $\mathbb{C}$ that $n = \omega$, i.e., that the coordinate changes are analytic. To get a unified theory of manifolds over various

fields, without code duplication, it is therefore necessary to have a single definition of C^n functions for $n \in \mathbb{N} \cup \{\infty, \omega\}$.

A suitable definition of C^n functions in `mathlib` should solve simultaneously these three difficulties.

First-order differential calculus has been implemented in `mathlib` by Avigad in 2019, following [HIH13]. From 2019 to 2025, the author has brought C^n functions to `mathlib`, fixing successively several design flaws until the current definition, which solves the three issues above. The definition is now stable. It has proven its robustness, as testified by the multiple applications notably to manifold theory but also to Fourier analysis and distributions.

The current paper serves two goals. It is an entry point to people willing to use C^n functions in `mathlib`, describing the main API. It is also a justification of the design, explaining where more naive approaches would fail through several mathematical examples highlighting unexpected subtleties. This can be interesting to mathematicians or to formalizers using other systems.

Remark 1.3. There is another standard definition of C^n functions in terms of partial derivatives $\partial^n f/\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}$ with $\alpha_1 + \cdots + \alpha_d = n$, for $f : \mathbb{R}^d \to F$. This definition does not work in infinite dimension, and depends on the fact that the source space is precisely of the form $\mathbb{R}^d$ (this could be avoided by specifying instead a finite basis of the source vector space, or by considering partial derivatives in all directions). These shortcomings make it worse, both mathematically and from a formalization perspective, than Definition 1.2, although it may seem appealing at first as it requires less algebra (notably, no discussion of spaces of multilinear maps). This definition would not allow the study of Banach manifolds, or of calculus in infinite-dimensional Hilbert spaces which is key in quantum mechanics.

Partial derivatives may be recovered from the full iterated derivative. Denoting by e_i the i-th basis vector, then by definition

$$\frac{\partial^n f(x)}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}} = D^n f(x)(\underbrace{e_1, \ldots, e_1}_{\alpha_1 \text{ terms}}, \underbrace{e_2, \ldots, e_2}_{\alpha_2 \text{ terms}}, \ldots, \underbrace{e_d, \ldots, e_d}_{\alpha_d \text{ terms}}).$$

We will not use this point of view in this paper, as it is notationally and combinatorially more cumbersome than the conceptual approach of Definition 1.2 while applying in more restricted situations.

Since there are no dependent types in Isabelle/HOL, it is complicated to work with the space of n-multilinear maps from E^n to F, for varying n, so Immler and Zhan have to resort to this kind of definition of higher smoothness, based on partial derivatives, in [IZ19].

Remark 1.4. There is an even simpler definition of C^n functions when the source space is the field $\Bbbk$ itself. In this case, the Fréchet derivative at a point is a continuous linear map from $\Bbbk$ to F. There is a canonical identification between $\mathcal{L}(\Bbbk, F)$ and F, mapping L to $L(1)$. Through this identification, we recover the elementary notion of the one-dimensional derivative of a function $f : \Bbbk \to F$, as a function $f' : \Bbbk \to F$. This process can be iterated as the types do not change, yielding the n-th one-dimensional derivative $f^{(n)} : \Bbbk \to F$. A function is C^n in this context if all functions $f^{(m)}$ for $m \leq n$ are well defined and continuous. In the Mathcomp Analysis Library for Rocq, the iterated derivatives of functions are currently available in this one-dimensional setting.

Remark 1.5. There are a lot of other scales of smooth-like functions. For instance, in $\mathbb{R}^n$, taking into account integrability properties of the function and its successive derivatives with respect to Lebesgue measure (and/or using Fourier transform), one can define the Sobolev and Besov spaces. One can also index the smoothness by a nonnegative real number, by adding Hölder controls on the iterated derivatives. One can also consider definitions in terms of iterated differences operators instead of iterated derivatives, which turn out to work better over ultrametric fields as advocated in [Sch84] (solving for instance issues such as Proposition 5.2 below). We will not explore these other classes in this paper, and focus only on the formalization of a class based on Definition 1.2 as this is the most fundamental in the literature.

2. First-order calculus

As in Isabelle/HOL, the derivatives of functions in `mathlib` are defined for any normed field, on domains. The main predicate is

```
def HasFDerivWithinAt (f : E → F) (f' : E →L[𝕜] F) (s : Set E)
  (x : E) :=
 (fun y ↦ f y - f x - f' (y - x)) =o[𝕜; 𝓝[s] x] (fun y ↦ y - x)
```

(The letter `F` in `HasFDerivWithinAt` stands for *Fréchet*, to distinguish it from the one-dimensional derivative discussed in Remark 1.4.) In this definition, $\Bbbk$ is a normed field, E and F are two normed vector spaces over $\Bbbk$, and $E \to L[\Bbbk]F$ denotes the space of continuous linear maps from E to F, and $\mathcal{N}[s]x$ is the filter of neighborhoods of x within the set s. This is the direct analogue of Definition 1.2, except that the field of scalars does not have to be $\mathbb{R}$, and moreover this definition is within a domain, i.e., we only require that $f(y) = f(x) + f'(y-x) + o(y-x)$ when y belongs to the set s. Using domains is important for the application to manifolds with boundary, but also for more elementary statements like the fundamental theorem of calculus, where one considers derivatives inside intervals.

(The definition is not written exactly like this in `mathlib` as it involves another predicate `HasFDerivAtFilter`, but it is definitionally equal to what is written above).

Then we have the predicate `DifferentiableWithinAt 𝕜 f s x` registering that there exists a continuous linear map f' such that `HasFDerivWithinAt f f' s x`. Such a value for the derivative is recorded in `fderivWithin 𝕜 f s x`. This function is total, i.e., it is also defined when the function is not differentiable, for ease of use. We give the value 0 to the derivative in this case by convention, and also because this choice of junk value ensures that many lemmas remain true even for non-differentiable functions. Here is the formal definition:

```
def fderivWithin (𝕜) (f : E → F) (s : Set E) (x : E) : E →L[𝕜] F :=
 if HasFDerivWithinAt f (0 : E →L[𝕜] F) s x
  then 0
 else if h : DifferentiableWithinAt 𝕜 f s x
  then Classical.choose h
 else 0
```

In general domains (contrary to open sets), the derivative is not unique, hence the definition resorts to `Classical.choose` to choose some value of the derivative. Using the value 0 as the derivative whenever possible, as above, is convenient in several contexts – for instance, it ensures that the derivative in a domain of a constant function is zero, while otherwise the derivative of a constant function within the empty set could be arbitrary for instance.

`mathlib` also contains specializations of these notions when s is the whole space, called respectively `HasFDerivAt f f' x` and `DifferentiableAt 𝕜 f x` and `fderiv 𝕜 f x`. It also contains predicates `DifferentiableOn 𝕜 f s` recording that f is differentiable within s at each point of s, and `Differentiable 𝕜 f` recording that f is differentiable everywhere. A comprehensive API is then given about these notions, including standard facts such that the differentiability of the sum or the composition of differentiable functions.

There is a subtlety in domains. It is true that, if f and g have derivatives respectively f' and g' at x within s, then $f+g$ has derivative $f'+g'$ at x within s. In the same way, if f and g are differentiable at x within s, then $f+g$ also is. However, even if f and g are differentiable at x within s, it is *not* true in general that

```
fderivWithin 𝕜 (f + g) s x = fderivWithin 𝕜 f s x + fderivWithin 𝕜 g s x
```

because of the lack of uniqueness of derivatives within sets. For instance, assume that $E = \mathbb{R}^2$ and $F = \mathbb{R}$, let L be the linear map sending (x_1, x_2) to x_1 and M the linear map sending (x_1, x_2) to $x_1 + x_2$. Take $s = \mathbb{R} \times \{0\}$. The function L admits itself as a derivative everywhere, and in particular along s, and the same holds for M. However, as L and M coincide on s, the map L also admits M as a derivative along s. It is therefore possible that the functions $f = g = L$ have an `fderivWithin` along s equal to L, while $f + g = 2L$ may have an `fderivWithin` along s equal to $2M$, violating the above equality.

It is sometimes useful to ensure that the above equality holds. For this, the domain should be nice enough to enforce uniqueness of derivatives. This is true when the tangent directions of s at x span a dense subset of the whole space. We define a predicate `UniqueDiffWithinAt 𝕜 s x` registering this property, as well as `UniqueDiffOn 𝕜 s` registering that s is a set of unique differentials around each of its points. With these definitions at hand, the fact that the derivative of the sum is the sum of the derivatives reads for instance

```
theorem fderivWithin_add {f g : E → F} {x : E} {s : Set E}
  (hxs : UniqueDiffWithinAt 𝕜 s x)
  (hf : DifferentiableWithinAt 𝕜 f s x)
  (hg : DifferentiableWithinAt 𝕜 g s x) :
  fderivWithin 𝕜 (f + g) s x = fderivWithin 𝕜 f s x + fderivWithin 𝕜 g s x
```

For whole derivatives (i.e., when s is the whole space), no further conditions are needed as the whole space (or open sets) are sets of unique differentials. This corresponds to the uniqueness of derivatives mentioned after Definition 1.1. More generally, convex sets with nonempty interior are also sets of unique differentials. This applies for instance to the half-space or the quadrant, which turns out to be important in manifold theory.

3. Analytic functions

Our framework for C^n functions has to contain analytic functions, as explained on Page 2. In this paragraph, we describe our formalization of analytic functions in `mathlib`, as a prerequisite to formalize C^n functions.

A one-dimensional function f is analytic around 0 if it can be written as a convergent sum $f(z) = \sum_n c_n z^n$. This generalizes readily: a function $f : \mathbb{R}^d \to \mathbb{R}$ is analytic at 0 if it can be written around 0 as

$$f(x_1, \dots, x_d) = \sum_\alpha c_\alpha x_1^{\alpha_1} \cdots x_d^{\alpha_d},$$

where the sum is over all multiindices $\alpha = (\alpha_1, \dots, \alpha_d)$ and is required to converge in norm on a neighborhood of 0 (equivalently, the coefficients c_α should satisfy a bound $|c_\alpha| \le CR^{\alpha_1+\cdots+\alpha_d}$). However, this definition is restricted to finite dimension. The correct general definition is the following (see e.g. [Muj86]):

Definition 3.1. *Let E and F be two normed vector spaces over a normed field $\Bbbk$. A function $f : E \to F$ is analytic at x if there exists a sequence p_n, where $p_n : E^n \to F$ is a continuous n-multilinear map satisfying $\|p_n\| \leqslant CR^n$ for some $C > 0$ and $R < \infty$, such that for all y close enough to 0 one has*

$$f(x+y) = \sum_n p_n(y, \dots, y). \tag{3.1}$$

In one dimension, $p_n(y^{(1)}, \dots, y^{(n)})$ is just $c_n y^{(1)} \cdots y^{(n)}$. In the finite-dimensional situation, the term $p_n(y, \dots, y)$ is the sum of the terms $c_\alpha y_1^{\alpha_1} \cdots y_d^{\alpha_d}$ over all multiindices with $\alpha_1 + \cdots + \alpha_d = n$.

To formalize this definition, we define first the type `E [×n]→L[𝕜] F` of continuous n-multilinear maps from E^n to F, with its canonical normed space structure over $\Bbbk$ (where E and F are normed spaces over $\Bbbk$). Let `FormalMultilinearSeries 𝕜 E F` be the space of sequences p_n with p_n a continuous n-multilinear map. We introduce a structure recording that a function admits a power series expansion as in Definition 3.1, on a ball of radius r intersected with a set s as we want definitions in domains:

```
structure HasFPowerSeriesWithinOnBall (f : E → F)
    (p : FormalMultilinearSeries 𝕜 E F) (s : Set E)
    (x : E) (r : ℝ≥0∞) : Prop where
  r_le : r ≤ p.radius
  r_pos : 0 < r
  hasSum : ∀ y, x + y ∈ insert x s → y ∈ ball (0 : E) r
    → HasSum (fun n ↦ p n (fun _ : Fin n ↦ y)) (f (x + y))
```

In this definition, ℝ≥0∞ is the type of extended nonnegative reals, i.e., $[0, +\infty]$, and `p.radius` is the radius on which the power series defined by p is converging. The condition `hasSum` requires the condition (3.1) for $x + y \in s \cup \{x\}$ instead of just s as the theory is very bad otherwise, and in interesting applications x belongs to s anyway.

Once this predicate is available, we say that f is analytic within s at x if there exist p and $r > 0$ such that f has a power series given by p on the ball of radius r. Formally,

```
def AnalyticWithinAt (𝕜) (f : E → F) (s : Set E) (x : E) :=
  ∃ p : FormalMultilinearSeries 𝕜 E F, ∃ r, HasFPowerSeriesWithinOnBall f p s x r
```

Note how this predicate is similar to `DifferentiableWithinAt 𝕜 f s x`, although it requires a much stronger smoothness property. In parallel to the differentiability situation, we introduce further predicates `AnalyticOn` recording that a function is analytic within s at every point of s, and `AnalyticAt` recording that a function is analytic at a point (which corresponds to s being the whole space). We also provide a comprehensive API for these notions.

The sequence p_n is not uniquely determined by f: any sequence p'_n with $p_n(y, \ldots, y) = p'_n(y, \ldots, y)$ will also do provided that its norm does not grow superexponentially.

Example 3.2. Let $\Bbbk$ be a normed field and let $f : \Bbbk^2 \to \Bbbk$ be given by $f(x_1, x_2) = x_1 x_2$. Let $p_n = 0$ for $n \neq 2$ and $p_2((x_1, x_2), (x'_1, x'_2)) = x_1 x'_2$. Then $f(y) = \sum_n p_n(y, \ldots, y)$, showing that f is analytic around 0. The sequence p'_n given by $p'_n = 0$ for $n \neq 2$ and $p'_2((x_1, x_2), (x'_1, x'_2)) = x'_1 x_2$ also satisfies $f(y) = \sum_n p'_n(y, \ldots, y)$.

If one wanted a canonical choice for p_n, one could try to enforce symmetry: in this example, one would set

$$p_2((x_1, x_2), (x'_1, x'_2)) = (x_1 x'_2 + x'_1 x_2)/2.$$

However, this is not possible if $\Bbbk$ is of characteristic 2, for instance $\Bbbk = \mathbb{F}_2((t))$. In this case, there is no symmetric representative, contrary to characteristic 0 where such a canonical choice is always possible.

If f is analytic around a point x, and y is small enough, then f is also analytic around $x + y$ if the target space is complete. In particular, the set of analyticity points of f is open. This follows from the following computation:

$$\begin{aligned} f(x + y + z) &= \sum_n p_n(y + z, \ldots, y + z) \\ &= \sum_n \sum_{I \subseteq \{0, \ldots, n-1\}} \sum p_n(w_1^I, \ldots, w_n^I) \end{aligned}$$

by multilinearity, where w_i^I is equal to z if $i \in I$, and y otherwise. Varying only the coordinates inside I, one can see the latter expression as a $\#I$-multilinear map. Grouping together the terms with the same number of variables, one gets $f(x+y+z) = \sum q_k(z, \ldots, z)$ where q_k is k-multilinear. To convert this into a rigorous proof, one should check the convergence of the series defining q_k, which follows from norm estimates and the completeness of the target space.

The main difficulty here is to write down things precisely, as is often the case when the paper proof involves a lot of ellipses. The norm controls are then straightforward albeit tedious.

If f is analytic at x (given by the series of the p_n), then f is also differentiable at x, with derivative equal to p_1 (modulo the identification between continuous 1-multilinear maps and continuous linear maps). This follows from the series expression and the fact that, for $n \geq 2$, then $p_n(y, \ldots, y) = O(\|y\|^n) = o(\|y\|)$ (together with uniform norm controls). This is formalized as

```
theorem HasFPowerSeriesAt.hasFDerivAt
  (h : HasFPowerSeriesAt f p x) :
  HasFDerivAt f
   (continuousMultilinearCurryFin1 𝕜 E F (p 1)) x

theorem AnalyticAt.differentiableAt
  (h : AnalyticAt 𝕜 f x) :
```

```
DifferentiableAt 𝕜 f x
```

Versions within a set are also given in the library.

Example 3.3. Here is an example showing that completeness of the target space is necessary for the openness of the set of analyticity points. Let $\ell^\infty(\mathbb{R})$ be the space of bounded real sequences, with the supremum norm. Define $f : \mathbb{R} \to \ell^\infty(\mathbb{R})$ as follows: for $|x| < 1$, let $f(x) = (1, x, x^2, \dots)$. Otherwise, let $f(x) = 0$.

The function f is analytic at 0. Indeed, for $|x| < 1$, we have $f(x) = \sum c_n x^n$ where $c_n \in \ell^\infty(\mathbb{R})$ is the sequence with a 1 at index n and zeroes elsewhere. As $\ell^\infty(\mathbb{R})$ is complete, if follows that f is analytic at any x with $|x| < 1$, and therefore differentiable. Its derivative is $\sum n c_n x^{n-1}$.

Let F be the vector subspace of $\ell^\infty(\mathbb{R})$ spanned by the c_n's and the values of $f(x)$ for $x \in \mathbb{R}$. This space is not complete any more. As f takes its values in F, one can define a new function $g : \mathbb{R} \to F$ equal to f. The expansion $g(x) = \sum c_n x^n$ is still true in F, so g is analytic at 0. However, we claim that g is not analytic at any other point x with $|x| < 1$, showing that the set of analyticity points of g is not open.

Assume by contradiction that g is analytic at x. Then g is differentiable at x. Therefore, the derivative $\sum_n n c_n x^{n-1}$ belongs to F. By definition of F, it is the sum of a finitely supported sequence, and a finite sum $\sum_i a_i (1, y_i, y_i^2, \dots)$. Looking at the n-th coefficient of the sequence, we get $n x^{n-1} = \sum_i a_i y_i^n$ for large n. Considering even values of n, we can even assume that all the y_i are nonnegative, replacing y_i by $-y_i$ if necessary. Then the growth rate of the right hand term is $c z^n$ where z is the largest of the y_i with a nonzero coefficient. This is a contradiction as this is supposed to grow like $n x^{n-1}$.

In this example, the function g is only differentiable at 0. Its derivative there is $x \mapsto c_1 x$. At other points, it is not differentiable. With our convention for the derivative, the derivative will then be zero at these other points. This implies that the derivative of g is not continuous at 0, albeit g is analytic there!

Among the standard results on analytic functions that we formalize, we prove that the composition of analytic functions is analytic, and that the local inverse of an analytic function is analytic, i.e., the analytic version of the inverse function theorem. Defining a sequence q_n for the inverse of a map f corresponding to a sequence p_n is easy formally, but checking that the growth of $\|q_n\|$ is not faster than any exponential requires a further argument. Standard textbooks use complex analysis and the Cauchy integral formula to control this growth. However, the development of analytic functions in `mathlib` comes *before* complex analysis (which uses a lot of material on analytic functions). Using it for the proof of the analytic inverse function theorem would therefore require a lot of intermingling between the two theories, harming the internal coherence of `mathlib`. Instead, we have devised a direct proof, of a more combinatorial nature and relying on a tricky inductive estimate originating in [Pös21].

All these constructions rely heavily on the finite sums and infinite sums implementations in `mathlib`, as well as results about completeness, which were already well developed before the start of this project.

4. C^n FUNCTIONS

4.1. The formal definition.

In this paragraph, we give the definition of C^n functions in `mathlib`. As the definition is more complicated than one might expect naively, we start with a discussion of several failed attempts to show the difficulties. We will only talk about C^n functions for $n \leq \infty$ at first, and add analytic functions later on.

Attempt 4.1. As a first attempt, we could follow literally Definition 1.2: define inductively the n-th iterated derivative of f within the set s, as the derivative within s of the $(n-1)$-th iterated derivative, and say that f is C^n within s if all these iterated derivatives are well defined and continuous up to stage n.

The non-uniqueness of derivatives in domains implies that this strategy fails: with this definition, one can not prove that continuous linear maps are C^n on domains, while this should hold for any reasonable definition.

For instance, consider the function $f : (x_1, x_2) \mapsto x_1$ defined on $\mathbb{R}^2$, and the domain $s = \mathbb{R} \times \{0\}$. Any linear map ℓ_α of the form $(u, v) \mapsto u + \alpha v$ is a derivative of f along s, as only the horizontal direction matters. It follows that `fderivWithin ℝ f s x`, as defined above using choice, might be any function $\ell_{\alpha(x)}$. If α is not continuous on s, then f would not even be C^1 on s.

This failed attempt shows that a reasonable definition should not be in terms of the specific derivative `fderivWithin ℝ f s x` (which might be badly behaved) but instead in terms of the existence of a nice sequence of derivatives. Using the type `FormalMultilinearSeries 𝕜 E F` of sequences p_n of n-multilinear maps, we record in `HasFTaylorSeriesUpToOn n f p s` the fact that p_0 coincides with f on s, and p_{m+1} is the derivative of p_m for any $m < n$, and p_n is continuous. This would make it possible to say that a function is C^n if it admits a Taylor series of order n (except that in reality we will again need to adapt this definition).

The formalized definition is the following:

```
structure HasFTaylorSeriesUpToOn
  (n : WithTop ℕ∞) (f : E → F)
  (p : E → FormalMultilinearSeries 𝕜 E F) (s : Set E) : Prop where
 zero_eq : ∀ x ∈ s, (p x 0).curry0 = f x
 fderivWithin : ∀ m : ℕ, m < n → ∀ x ∈ s,
  HasFDerivWithinAt (p · m) (p x m.succ).curryLeft s x
 cont : ∀ m : ℕ, m ⩽ n → ContinuousOn (p · m) s
```

In this definition, `WithTop ℕ∞` is the set $\mathbb{N}$ with two added elements ∞ and ω. We include these two added elements to anticipate over the definition of C^n for $n \in \mathbb{N} \cup \{\infty, \omega\}$. The `F` in the name stands for "Fréchet", to emphasize that this is not a one-dimensional notion. The property `zero_eq` essentially requires that f and $p_0(x)$ coincide on s. However, $f(x)$ belongs to F while $p_0(x)$ is a function from E^0 to F, so they can not coincide formally: to get a sensible statement, one needs to add the canonical identification between F and $E^0 \to F$, which we may see as a curryfication step. In the same way, the property `fderivWithin` requires essentially that $p_{m+1}(x)$ is a derivative of p_m at x, except that p_{m+1} is a multilinear map $E^{m+1} \to F$ while the derivative of p_m is a linear map from E to multilinear maps from E^m to F. Again, these two sets of functions are canonically identified through a curryfication step, written as `curryLeft` in the definition. Finally, we should also require the continuity of p_n, except that it does not make sense for $n = \infty$ or ω. Therefore, `cont` requires instead continuity of all p_m for $m \leq n$, although the new information is only for the last term if there is one: by `fderivWithin` we already know that the other terms are differentiable, hence continuous.

Remark 4.2. Formal multilinear series, i.e., sequences (p_n) of continuous n-multilinear maps, show up both in the definition of analytic functions as $f(x+y) = \sum p_n(y, \dots, y)$ in Definition 3.1, and in sequences of iterated derivatives in Taylor series. These two uses of the same formal object do *not* correspond to each other. For instance, consider the real function $f(x) = 1/(1-x)$ around 0. It is a analytic at 0, with expansion $f(x) = \sum x^n$, so $p_n(y, \dots, y) = y^n$. The n-th derivative of f at 0, however, is $D^n f(0)(v_1, \dots, v_n) = n! v_1 \cdots v_n$ (corresponding to the statement in terms of 1-dimensional derivatives that $f^{(n)}(0) = n!$). There is a discrepancy of $n!$ between the analytic function point of view and the derivatives point of view. This is not surprising given the standard expansion of a one-dimensional real analytic function around 0,

$$f(y) = \sum \frac{f^{(n)}(0)}{n!} y^n.$$

The coefficient $n!$ here has a nice combinatorial interpretation as the cardinality of the group of permutations of $\{1, \dots, n\}$, see (5.2).

Attempt 4.3. As a second attempt, we could say that a function is C^n (for $n \in \mathbb{N} \cup \{\infty\}$) on a set s if there exists a sequence (p_m) which is a Taylor series for f up to n on s in the above sense.

The issue with this definition is locality, both in space and in regularity. A good notion of C^n function should satisfy the following: assume that, for all $x \in s$, there is a neighborhood u of x in s on which f is C^n. Then f is C^n on s. Also, if a function is C^n on s for all natural number n, then it should be C^∞. The above definition a priori fails these two properties, as we will explain now.s

If f is C^n in this sense on a neighborhood u of x in s, one gets a corresponding sequence of multilinear maps $p_m^{(u)}$, indexed by u, giving a Taylor series for f on u. If these were compatible in the sense that $p_m^{(u)} = p_m^{(v)}$ on $u \cap v$, one could glue them together for f on the whole s. However, due to the lack of uniqueness of derivatives, there is no guarantee of compatibility. In a finite-dimensional real vector space, one could instead extract a locally finite covering of the u's, then use a smooth partition of unity ρ_u subordinate to this locally finite covering, and form the (locally finite) sum $\sum_u \rho_u p_m^{(u)}$. This would be a Taylor series for f on the whole set s. However, the existence of smooth partitions of unity is not always true in infinite dimension, or on other fields. For instance, there is no complex-differentiable partition of unity on $\mathbb{C}$ as a complex-differentiable function with compact support vanishes (as it is analytic).

Our solution to this issue is to bake in locality into the definition: we will say that f is C^n within s at x (for $n \in \mathbb{N} \cup \{\infty\}$) if, for every natural number $n' \leq n$, there exist a neighborhood u of x within s and a formal multilinear series (p_m) such that f admits (p_m) as a Taylor series on v up to order n'. Then, we will say that f is C^n on s if it is C^n within s at every point of s. By design, this definition is local both in space and in regularity.

Let us now integrate analytic functions into our definition of C^n functions.

Attempt 4.4. As a third attempt, let us say that f is C^n within s at x (for $n \in \mathbb{N} \cup \{\infty\}$) using the above local definition, and that it is C^ω within s at x if it is analytic within s at x.

With this definition, however, it is not true that if f is C^n and $m \leq n$ then f is C^m, while this is a property we definitely expect of a scale of smooth functions. This is due to the bad behavior of analytic functions on non-complete spaces. For instance, the function g of Example 3.3 is analytic at 0 but it is not C^1 at 0, as explained at the end of the example.

A solution would be to require all our target spaces to be complete, as is for instance done in [Bou71]. However, this is unnecessarily restrictive: most of the theory can be built without completeness, and completeness should only be brought as an assumption in theorems that really need it. Instead, our strategy is to require that C^ω functions are analytic, as well as all their derivatives. In this way, we recover monotonicity of the scale of smooth functions all the way up to ω.

We can now give the formalized definition of C^n functions used in `mathlib`. We introduce a predicate `ContDiffWithinAt 𝕜 n f s x` registering that f is C^n (over the field 𝕜) within s at x, defined as follows:

```
def ContDiffWithinAt
  (𝕜) (n : WithTop ℕ∞) (f : E → F) (s : Set E) (x : E) : Prop :=
 match n with
 | (n : ℕ∞) => ∀ m : ℕ, m ≤ n → ∃ u ∈ 𝓝[insert x s] x,
    ∃ p : E → FormalMultilinearSeries 𝕜 E F, HasFTaylorSeriesUpToOn m f p u
 | ω => ∃ u ∈ 𝓝[insert x s] x,
    ∃ p : E → FormalMultilinearSeries 𝕜 E F,
    HasFTaylorSeriesUpToOn ω f p u ∧
    ∀ i, AnalyticOn 𝕜 (fun x ↦ p x i) u
```

If the regularity n belongs to $\mathbb{N} \cup \{\infty\}$, then for all integer $m \leq n$ we require the existence of a neighborhood u of x within $s \cup \{x\}$ on which f admits a Taylor series up to order m. This corresponds to the above discussion, except that we use $s \cup \{x\}$ instead of s: this avoids pathologies where the Taylor series would be discontinuous at x. For $n = \omega$, we require the existence of a neighborhood u of x within s on which f admits a Taylor series at infinite order, made of analytic functions.

Once we have this predicate, we define a version not localized to a set, called `ContDiffAt` (obtained by taking for s the whole space in the above definition), a version asking that f is C^n within s at all points of s called `ContDiffOn`, and a version in the whole space called `ContDiff`. These are the perfect analogues of the corresponding definitions for differentiable functions and for analytic functions: such a coherence is very helpful to the users of the library. The lemmas for these three notions also follow a completely identical naming scheme.

Let us now give a few basic statements showing that the definition above lives up to our promises.

The scale of C^n functions is monotone with respect to n:

```
theorem ContDiffWithinAt.of_le
  (h : ContDiffWithinAt 𝕜 n f s x) (hmn : m ≤ n) :
  ContDiffWithinAt 𝕜 m f s x
```

In particular, a C^ω function is C^∞.

In a complete space, C^ω and analytic coincide:

```
theorem contDiffWithinAt_omega_iff_analyticWithinAt
  [CompleteSpace F] :
  ContDiffWithinAt 𝕜 ω f s x ↔ AnalyticWithinAt 𝕜 f s x
```

For $n \neq \infty$, a function is C^n within a set at a point if and only if there exists a neighborhood on which it has a Taylor series at order n, and moreover if $n = \omega$ all terms in this Taylor series should be analytic:

```
lemma contDiffWithinAt_iff_of_ne_infty (hn : n ≠ ∞) :
  ContDiffWithinAt 𝕜 n f s x ↔ ∃ u ∈ 𝓝[insert x s] x,
   ∃ p : E → FormalMultilinearSeries 𝕜 E F, HasFTaylorSeriesUpToOn n f p u ∧
    (n = ω → ∀ i, AnalyticOn 𝕜 (fun x ↦ p x i) u)
```

Note that this is not true for $n = \infty$: consider $f : \mathbb{R} \to \mathbb{R}$ which is C^n and not better on $(-1/n, 1/n)$. Then it is C^∞ at 0, but not C^∞ on a neighborhood of 0.

For $n \neq \infty$, a function is C^{n+1} on a domain iff locally, it has a derivative which is C^n (and moreover the function is analytic when $n = \omega$).

```
theorem contDiffWithinAt_succ_iff_hasFDerivWithinAt
  (hn : n ≠ ∞) :
  ContDiffWithinAt 𝕜 (n + 1) f s x ↔ ∃ u ∈ 𝓝[insert x s] x,
   (n = ω → AnalyticOn 𝕜 f u) ∧ ∃ f' : E → E →L[𝕜] F,
   (∀ x ∈ u, HasFDerivWithinAt f (f' x) u x) ∧ ContDiffWithinAt 𝕜 n f' u x
```

4.2. **Operations on functions respect smoothness.** There is a tension between our definition and standard proofs of properties of C^n functions. Here is the standard proof that the sum of two C^n functions is C^n, based on Definition 1.2.

Proposition 4.5. *Let E, F be two real normed spaces and $f, g : E \to F$ be two C^n functions for $n \in \mathbb{N}$. Then $f + g$ is C^n.*

Proof. We argue by induction on n. For $n = 0$, the result is standard. For $n > 0$, according to Definition 1.2, we should check that $f + g$ is differentiable, which is standard, and that $D(f + g)$ is C^{n-1}. As $D(f + g) = Df + Dg$, this follows from the inductive assumption applied to the functions Df and Dg from E to $\mathcal{L}(E, F)$. □

Our definition of C^n functions in `mathlib` is not recursive, so the above proof can not be adapted directly. However, we have another recursive characterization stated above in `contDiffWithinAt_succ_iff_hasFDerivWithinAt`, that we could try to use. This approach has two issues.

First, it does not work for C^ω functions: a function is C^ω if it is analytic and has a derivative which is C^ω – this can not be used in an inductive-like argument. Second, there are universe polymorphism issues. The proof above applies the inductive assumption to another pair of functions, Df and Dg, with a different target space, $\mathcal{L}(E, F)$. Therefore, a proper formulation of the inductive assumption would be: for all real normed spaces E and F and all pairs of C^n functions f, g from E to F, the sum $f + g$ is C^n. To avoid Russell-like paradoxes, in the dependent type theory of Lean, spaces live in a *universe*, which can not be quantified in statements or proofs: one can not write "for all universes u, for all spaces in the universe u, then ...” In particular, for an induction to make sense, the spaces featured in the induction should live in a fixed universe. If E is in a universe u and F is in a universe v, then $\mathcal{L}(E, F)$ lives in another universe $\max(u, v)$, which means that our induction will not work correctly. This issue can be worked around as follows. First prove a less polymorphic statement, in which E and F belong to the same universe u (as well as all the spaces of multilinear maps that show up in the proof, enabling the induction). Then reduce the general case to the specific case: lift E and F to spaces E' and F' in a common universe, together with C^n lifted functions f' and g', then argue that $f' + g'$ is C^n using the less polymorphic statement, and finally deduce that $f + g$ is C^n as being C^n is invariant under continuous linear isomorphisms.

Given these two difficulties, and especially the first one which can not be worked around, it follows that the formalized proof should probably not follow the inductive classical proof above, and instead be more in line with the definition choice we have made. Here is another paper proof of Proposition 4.5, but closer to our formal definition.

Another proof of Proposition 4.5. As f and g are C^n, they admit respective Taylor series p and q up to order n. Then one checks readily that $p + q$ is a Taylor series up to order n for $f + g$, proving that $f + g$ is C^n. □

There is no inductive character to this proof, which means that it is straightforward to formalize, to get the following:

```
theorem ContDiffWithinAt.add
  (hf : ContDiffWithinAt 𝕜 n f s x)
  (hg : ContDiffWithinAt 𝕜 n g s x) :
  ContDiffWithinAt 𝕜 n (fun x ↦ f x + g x) s x
```

A lesson here is that, instead of proving inductively that functions are C^n, it works better with our definition to *give a formula* for the successive derivatives, as an operation on the Taylor series of the original functions. For the sum of functions, this is very easy (the Taylor series of the sum is the sum of the Taylor series), but this creates a difficulty for the composition.

Theorem 4.6. *Let E, F, G be real normed spaces. Consider $f : E \to F$ and $g : F \to G$ two C^n functions. Then the composition $g \circ f$ is C^n.*

Here is the standard textbook inductive proof.

Proof. The statement is easy for $n = 0$. For $n > 0$, the function $g \circ f$ is differentiable as the composition of differentiable functions, and its derivative is given by $D(g \circ f)(x)(v) = Dg(f(x))(Df(x)(v))$. We should show that this is a C^{n-1} function of x.

Let $M : \mathcal{L}(E, F) \times \mathcal{L}(F, G) \to \mathcal{L}(E, G)$ be the composition of linear maps. This is a continuous bilinear map, hence C^∞. Let $\Phi : E \to \mathcal{L}(E, F) \times \mathcal{L}(F, G)$ be $(Df, (Dg) \circ f)$. As f is C^n, then Df is C^{n-1}. Moreover, $(Dg) \circ f$ is C^{n-1} using our inductive assumption, as the composition of two C^{n-1} functions. It follows that Φ is C^{n-1}. By construction, $D(g \circ f) = M \circ \Phi$. It is the composition of two C^{n-1} functions, therefore C^{n-1} again by our inductive assumption. □

It is not obvious from this proof how to give a formula for the derivative of $g \circ f$, and the formula would probably be very complicated. For instance, the second derivative is

$$\begin{aligned} D^2(g \circ f)(x)(v_0, v_1) &= D^2 g(f(x))(Df(x)v_0, Df(x)v_1) \\ &\quad + Dg(f(x))(D^2 f(x)(v_0, v_1)). \end{aligned}$$

In a previous `mathlib` implementation of C^n functions, in which analytic functions were not yet integrated in the smooth hierarchy, the fact that the composition of C^n functions is C^n was proved following the above inductive argument (and dealing with the universe polymorphism issue thanks to a lifting step, as explained above). This made it possible to avoid the complicated formula for the n-th derivative of a composition.

In the refactor integrating analytic functions in the smooth hierarchy, the smoothness of composition of functions has been the most difficult point: we had to switch proofs since the inductive argument does not make sense for C^ω functions. Instead, we state and prove the formula for the Taylor series of a composition, called the Faà di Bruno formula. This formula is usually given for functions of one variable, or using partial derivatives for functions of finitely many variables, but there is a general formula in terms of Fréchet derivatives which we will now describe. It is expressed in terms of partitions I of $\{0, ..., n-1\}$. For such a partition, denote by k its number of parts, write the parts as $I_0, \ldots, I_{k-1}$ ordered so that $\max I_0 < ... < \max I_{k-1}$, and let i_m be the number of elements of I_m. Then

$$D^n(g \circ f)(x)(v_0, \ldots, v_{n-1}) = \sum_I D^k g(f(x))(D^{i_0} f(x)(v_{I_0}), ..., D^{i_{k-1}} f(x)(v_{I_{k-1}}))$$

where the sum is over all partitions of $\{0, \ldots, n-1\}$ and by v_{I_m} we mean the family of vectors (v_i) with indices i in I_m.

The formula is straightforward to prove by induction. Differentiating

$$D^k g(f(x))(D^{i_0} f(x)(v_{I_0}), \ldots, D^{i_{k-1}} f(x)(v_{I_{k-1}})),$$

we get a sum with $k+1$ terms where one differentiates either $D^k g(f(x))$ or one of the $D^{i_m} f(x)$, amounting to adding to the partition I either a new atom $\{-1\}$ to its left, or extending I_m by adding -1 to it. In this way, one obtains bijectively all partitions of $\{-1, \ldots, n-1\}$, and the proof can go on (up to relabeling).

The formalization of this formula is rather intricate. Even writing down things precisely is hard, as indicated by the number of ellipses in the above formula. The most technical part is to show that extending partitions of $\{0, \ldots, n-1\}$ by the processes described above, one obtains bijectively all partitions of $\{-1, \ldots, n-1\}$. Overall, the formalization of the Faà di Bruno formula takes slightly more than 1000 lines.

Once this formula is available, the proof that the composition of C^n functions is C^n is easy, as the formula tells how to construct a Taylor series for $g \circ f$ in terms of Taylor series for g and f. For C^ω functions, one should additionally use that the composition of analytic functions is analytic, a fact which has already been discussed in Section 3.

Closely related to composition is inversion. The inverse function theorem states that, if a function between complete spaces is C^n at a point x_0 with invertible derivative, then it admits a local inverse which is also C^n. There are two parts in this theorem. First, if f is C^1 at x_0 with invertible derivative, then it admits a local inverse g. Second, if in fact f is C^n, then g is also automatically C^n. The two parts are essentially independent.

The deeper one is the first one, constructing the inverse by a suitable application of the Banach contraction principle. The C^1 assumption is not the correct one over a general field: one should instead require that f is strictly differentiable at x_0, i.e., $f(y)-f(z)-Df(x_0)(y-z)=o(y-z)$ as y and z tend to x_0. This follows from a C^1 assumption over $\mathbb{R}$ or $\mathbb{C}$ (by the mean value inequality), but not over general fields. However, strict differentiability follows from analyticity, which is the right smoothness condition to consider over general fields anyway as further discussed in Section 5.

For the second part, one can check the regularity inductively: as $f \circ g = \mathrm{id}$, one gets $Df(g(x))Dg(x) = \mathrm{id}$, so $Dg(x) = Df(g(x))^{-1}$. This formula makes it possible to transfer a C^n assumption on g to a C^n assumption on Dg, deducing that g is C^{n+1} and proceeding inductively. Note that this argument deduces that g is C^{n+1} from the fact that it is C^n. The spaces or the maps do not change over the induction, so there is no formalization difficulty here. The analytic case has to be treated separately as one needs additional growth controls, but this has already been discussed in Section 3. Altogether, we get a generic statement of the inverse function theorem, working both for $n \neq \omega$ and $n = \omega$ in a unified way.

The first step is formalized by constructing a local inverse from a strictly differentiable map with invertible derivative. This is phrased in `mathlib` with `PartialHomeomorph`, which contains the data of a map (here f) and another map, which are locally inverse to each other.

```
def HasStrictFDerivAt.toPartialHomeomorph
  (f : E → F) {f' : E ≃L[𝕜] F} [CompleteSpace E]
  (hf : HasStrictFDerivAt f f' a) :
  PartialHomeomorph E F
```

Here, `E ≃L[𝕜] F` is the space of continuous linear bijections between E and F. Therefore, our assumption `hf` is a way of saying that the derivative of f is invertible.

The second step is formalized by saying that if two functions are locally inverse to each other, then the second inherits the regularity properties of the first:

```
theorem PartialHomeomorph.contDiffAt_symm
  [CompleteSpace E] (f : PartialHomeomorph E F)
  {f₀' : E ≃L[𝕜] F} {a : F} (ha : a ∈ f.target)
  (hf₀' : HasFDerivAt f f₀' (f.symm a))
  (hf : ContDiffAt 𝕜 n f (f.symm a)) :
  ContDiffAt 𝕜 n f.symm a
```

This can be applied to the partial homeomorphism `HasStrictFDerivAt.toPartialHomeomorph f hf` that has been constructed in the first step.

4.3. **Iterated derivatives.** To circumvent the difficulties coming from the non-uniqueness of derivatives, we have defined C^n functions via the existence of a nice Taylor series. However, as for `fderivWithin` and `fderiv`, it is often useful to have available *some* choice of an iterated derivative – although in general one will only be able to prove that it behaves well under a unique differentials assumption on the domain. There are two possible choices here: one may define the $(n+1)$-th iterated derivative either as the derivative of the n-th derivative, or as the n-th derivative of the derivative. For type-theoretic reasons, the n-th derivative of the derivative can not really be used: if it were an inductive definition over n, it would also need to generalize the spaces as to get the $n+1$-th derivative of $f : E \to F$ one would need the n-th derivative of $Df : E \to \mathcal{L}(E, F)$. Since the universe of $\mathcal{L}(E, F)$ is usually not the same as that of F, this can not be done in a plain inductive definition in Lean. Therefore, we define the $(n+1)$-th derivative as the derivative of the n-th derivative (modulo currying). This is in line with our definition of C^n functions using sequences of multilinear maps p_n, where p_{n+1} has to be a derivative of p_n.

The formal definition is the following:

```
def iteratedFDerivWithin (𝕜) (n : ℕ) (f : E → F) (s : Set E) :
  E → E[×n]→L[𝕜] F :=
 Nat.recOn n
  (fun x ↦ ContinuousMultilinearMap.uncurry0 𝕜 E (f x))
  (fun _ rec x ↦ ContinuousLinearMap.uncurryLeft
   (fderivWithin 𝕜 rec s x))
```

This definition is written using the bare recursor `Nat.recOn` instead of versions with more syntactic sugar because of the position of n in the arguments and of difficulties of the elaborator with the complicated dependent type `E[×n]→L[𝕜] F`. Its content is however what we expect: for $n = 0$, use the function seen as a 0-multilinear map, and for $n + 1$ use the derivative of the n-th derivative, seen as an $(n + 1)$-multilinear map.

We also define `iteratedFDeriv`, corresponding to taking for s the whole space.

In domains with unique differentials (including the whole space, open sets, or convex sets with nonempty interior), if a function has a Taylor series up to order n then this Taylor series has to coincide with `iteratedFDerivWithin`, by uniqueness. Therefore, our proofs that the sum or composition of C^n functions are C^n, giving formulas for the Taylor series, yield also formulas for iterated derivatives. For instance, for the addition, we get

```
theorem iteratedFDerivWithin_add_apply
  (hf : ContDiffWithinAt 𝕜 i f s x)
  (hg : ContDiffWithinAt 𝕜 i g s x)
  (hu : UniqueDiffOn 𝕜 s) (hx : x ∈ s) :
  iteratedFDerivWithin 𝕜 i (f + g) s x =
   iteratedFDerivWithin 𝕜 i f s x
   + iteratedFDerivWithin 𝕜 i g s x
```

When we defined C^n functions, we had to take care of locality to avoid the non-uniqueness of derivatives, as discussed in Attempt 4.3. In domains with unique differentials, the issue disappears: different Taylor series around different points have to coincide locally, and can therefore be glued together. In other words, they coincide with a global object, which is `iteratedFDerivWithin`. Let `ftaylorSeriesWithin 𝕜 f s` be the Taylor series formed using the sequence of iterated derivatives. We get

```
theorem ContDiffOn.ftaylorSeriesWithin
  (h : ContDiffOn 𝕜 n f s) (hs : UniqueDiffOn 𝕜 s) :
  HasFTaylorSeriesUpToOn n f
   (ftaylorSeriesWithin 𝕜 f s) s
```

We have mentioned above that for the inductive definition of `iteratedFDerivWithin`, we had to write it as the derivative of the iterated derivative, for type theoretic reasons.

However, once the definition is available, one can prove that it coincides with the other natural definition, i.e., it is also the iterated derivative of the derivative (modulo currying):

```
theorem iteratedFDerivWithin_succ_eq_comp_right
  (hs : UniqueDiffOn 𝕜 s) (hx : x ∈ s) :
  iteratedFDerivWithin 𝕜 (n + 1) f s x =
  (continuousMultilinearCurryRightEquiv' 𝕜 n E F).symm
   (iteratedFDerivWithin 𝕜 n (fderivWithin 𝕜 f s) s x)
```

This requires the domain to have unique differentials, as otherwise the non-uniqueness of derivatives will break the result. Note that there is no differentiability assumption in this result: this is made possible by our choice of junk value 0 for the derivative of a non-differentiable function.

Remark 4.7. The formalization of the properties of C^n functions contains many curryfication or uncurryfication steps to identify for instance $\mathcal{L}(E, \mathcal{L}(E, F))$ with continuous bilinear maps from $E \times E$ to F. In textbooks such as [Car71], these identifications are discussed at the beginning, and they are then used implicitly everywhere. There is a subtlety here, though. Let $D(Df)$ denote the derivative of the derivative of f, as an element of $\mathcal{L}(E, \mathcal{L}(E, F))$, and $D^2 f$ its continuous bilinear map version. When we say that a function is C^3, are we saying that $D(Df)$ has a derivative which is continuous, or that $D^2 f$ has a derivative which is continuous? And do we get the same results by uncurrying those two derivatives? The answer to these two questions is yes, fortunately. This comes from the fact that the curryfication process is itself a continuous linear bijection between two vector spaces of functions, and continuous linear bijections respect differentiability and commute with taking derivatives. With our choice of junk values that non-differentiable functions have zero derivatives, composing with a continuous linear bijection and taking the derivative commute even for non-differentiable functions, which is extremely convenient in the formalization process. Note however that this discussion is only true for functions on domains with unique differentials: otherwise, the non-uniqueness of derivatives breaks this commutation, which has to be accounted for in the formalization.

5. Symmetry of iterated derivatives

An important property of the second derivative in real vector space is its symmetry $D^2 f(x)(v, w) = D^2 f(x)(w, v)$. Here is a precise version with weak assumptions, see [Car71, Theorem 5.1.1]:

Theorem 5.1. *Let E, F be two real normed vector spaces and let $f : E \to F$. Assume that f is differentiable on a neighborhood of a point x, and that Df is differentiable at x. Then $D^2 f(x)(v, w) = D^2 f(x)(w, v)$ for all vectors v, w.*

This result follows from the fact that $(f(x + tv + tw) - f(x + tv) - f(x + tw) + f(x))/t^2$ converges to $D^2 f(x)(v, w)$ as t tends to 0. Since the expression on the left is symmetric in v, w, it follows that the limit also is.

This result is fundamental for many constructions in geometry. For instance, define the Lie bracket of two vector fields $V, W : E \to E$ by

$$[V, W](x) = DW(x)(V(x)) - DV(x)(W(x)).$$

It encodes the commutation default between the flows of V and W. This definition is invariant under local diffeomorphisms, hence it extends to manifolds. To check the invariance, an order 2 term comes out, of the form

$$D^2 f(x)(V(x), W(x)) - D^2 f(x)(W(x), V(x)). \tag{5.1}$$

It vanishes thanks to the symmetry of the second derivative. In the same way, the relation $\mathrm{d} \circ \mathrm{d} = 0$ on differential forms ultimately boils down to a computation relying on the symmetry of the iterated derivative.

We have formalized the symmetry of the second derivative, in a slightly more general version as we allow convex domains:

```
theorem Convex.second_derivative_within_at_symmetric
  (s_conv : Convex ℝ s) (hne : (interior s).Nonempty)
  (hf : ∀ x ∈ interior s, HasFDerivAt f (f' x) x) (xs : x ∈ s)
  (hx : HasFDerivWithinAt f' f'' (interior s) x) (v w : E) :
  f'' v w = f'' w v
```

This extends readily from real vector spaces to complex vector spaces, since one can see a complex vector space as a real vector space. To cover also p-adic manifolds, we would like to extend this property to other normed fields, but unfortunately the symmetry of the second derivative fails in general: the proof over the reals uses the mean value inequality, which does not hold in discontinuous settings. We will describe a counterexample, which is essentially [Sch84, Corollary 84.2]. The following proposition shows that the derivative of a function on $\mathbb{Q}_p^2$ can be essentially arbitrary, from which one can get non-symmetric second derivatives. For simplicity, we work in the open subset $\mathbb{Z}_p$ of $\mathbb{Q}_p$, where p is any fixed prime number. The reader not familiar with p-adic numbers may safely skip this argument.

Proposition 5.2. *Let p be a prime number. For $x, y \in \mathbb{Z}_p^2$, let $M(x, y)$ be a matrix from $\mathbb{Q}_p^2$ to $\mathbb{Q}_p$ depending continuously on (x, y). Then there exists $f : \mathbb{Z}_p^2 \to \mathbb{Q}_p$ a function of class C^1 whose derivative is M.*

Note that the analogous result is completely false on $\mathbb{R}$, since the integral of the derivative along a loop from 0 to 0 must be zero, so everything is much more rigid.

The above proposition allows us to construct a counterexample to the symmetry of second derivatives. Let $M(x, y) = (0, x)$. Then the corresponding function f has M as its derivative, which is C^∞, so it is itself C^∞. Furthermore, $\partial f/\partial x = 0$, so $\partial^2 f/\partial y \partial x = 0$, while $\partial f/\partial y = x$, so $\partial^2 f/\partial x \partial y = 1$.

Proof. A point in $\mathbb{Z}_p$ is determined by its successive reductions in $\mathbb{Z}/p^n\mathbb{Z}$: at each stage, we refine by adding the data of the $n + 1$-th digit. Equivalently, $\mathbb{Z}_p$ is a union of p^n balls of radius p^{-n}, each ball being the union of p balls of radius $p^{-(n+1)}$. The same applies in $\mathbb{Z}_p^2$, except that there are p^{2n} balls of radius p^{-n}. Let us choose a center for each ball (the canonical choice is to take the point of the ball whose subsequent digits are all zero), and let C_n be the set of centers of balls of depth n, with $C_n \subseteq C_{n+1}$.

We define f recursively on C_n, starting from $f(0) = 0$. Suppose f is defined on C_n. For $b \in C_{n+1}$, let $a \in C_n$ be in the same ball of radius p^{-n}. We then set $f(b) = f(a) + M(a)\cdot(b-a)$. Note that for $a = b$, both sides give the same formula, so this inductive definition is consistent. Furthermore, $\|f(b) - f(a)\| \leqslant K\|b - a\| \leqslant Kp^{-n}$. The summability of this series ensures that the function f thus defined on the increasing union of the C_n is uniformly continuous. Therefore, it extends uniquely to a function on $\mathbb{Z}_p^2$, which we again denote by f.

It remains to be shown that f is differentiable at each point a, with derivative $M(a)$. Let $a_n \in C_n$ be the sequence of centers of the balls containing a, and similarly for a point b. Then

$$f(b) - f(a) = \sum M(b_n) \cdot (b_{n+1} - b_n) - \sum M(a_n)(a_{n+1} - a_n).$$

Suppose that $\|b - a\| = p^{-(N+1)}$. Then $a_n = b_n$ for $n \leq N$, so that we have

$$f(b) - f(a) = \sum_{n \geq N} M(b_n) \cdot (b_{n+1} - b_n) - \sum_{n \geq N} M(a_n)(a_{n+1} - a_n).$$

Let $\varepsilon > 0$. If N is large enough, the two matrices $M(b_n)$ and $M(a_n)$ are ε-close to $M(a)$, by continuity of M. Therefore, $f(b) - f(a)$ equals

$$\begin{aligned}
&M(a) \sum_{n \geq N} (b_{n+1} - b_n) - M(a) \sum_{n \geq N} (a_{n+1} - a_n) \pm 2\varepsilon \sum_{n \geq N} p^{-n} \\
&= M(a)(b - b_N) - M(a)(a - a_N) \pm 2\varepsilon p^{-N}/(1 - p^{-1}) \\
&= M(a)(b - a) \pm 2\varepsilon \|b - a\| p/(1 - p^{-1}).
\end{aligned}$$

We have shown that, as soon as b is close enough to a, we have for $C = 2p/(1 - p^{-1})$ the estimate

$$\|f(b) - f(a) - M(a)(b - a)\| \leqslant C\varepsilon \|b - a\|.$$

This is the definition of differentiability at a with derivative $M(a)$. □

The construction is sufficiently explicit that we can write everything down. For the counterexample to the symmetry of second derivatives, with $M(x, y) = (0, x)$, the function f reads

$$f\Big(\sum x_n p^n, \sum y_n p^n\Big) = \sum_{k<l} x_k y_l p^{k+l}.$$

The above pathology can not happen for analytic maps: if f is locally given by a series $f(x+y) = \sum_n p_n(y, \ldots, y)$ as in Definition 3.1, then one can compute the iterated derivative of f as follows: one has

$$D^n f(x)(v_1, \ldots, v_n) = \sum_{\sigma \in \mathfrak{S}_n} p_n(v_{\sigma(1)}, \ldots, v_{\sigma(n)}) \tag{5.2}$$

where $\mathfrak{S}_n$ denotes the set of permutations of $\{1, \ldots, n\}$. This expression is obviously symmetric in the v_i. In particular,

$$D^2 f(x)(v_1, v_2) = p_2(v_1, v_2) + p_2(v_2, v_1).$$

It follows from this discussion that symmetry of second derivatives of C^n functions holds when $n \geq 2$ if the field is $\mathbb{R}$ or $\mathbb{C}$, and for $n = \omega$ otherwise. All advanced calculus on vector fields or differential forms is only possible under such smoothness conditions. This is the reason why, over fields different from $\mathbb{R}$ or $\mathbb{C}$, Bourbaki only considers analytic manifolds in [Bou71].

We formalize the above formula for the iterated derivative of an analytic function:

```
theorem HasFPowerSeriesWithinOnBall.iteratedFDerivWithin_eq_sum
  (h : HasFPowerSeriesWithinOnBall f p s x r)
  (h' : AnalyticOn 𝕜 f s)
  (hs : UniqueDiffOn 𝕜 s) (hx : x ∈ s) (v : Fin n → E) :
  iteratedFDerivWithin 𝕜 n f s x v
   = ∑ σ : Perm (Fin n), p n (fun i ↦ v (σ i))
```

Let us now state that the second derivative is symmetric under a smoothness condition that depends on the field. Let `minSmoothness 𝕜 n` be equal to n if the field is isomorphic as a normed field to $\mathbb{R}$ or $\mathbb{C}$, and ω otherwise. The general formalized version of the second derivative symmetry is the following:

```
theorem ContDiffAt.isSymmSndFDerivAt (v w : E)
  (hf : ContDiffAt 𝕜 n f x) (hn : minSmoothness 𝕜 2 ≤ n) :
  fderiv 𝕜 (fderiv 𝕜 f) x v w = fderiv 𝕜 (fderiv 𝕜 f) x w v
```

This design choice with a function `minSmoothness` makes it possible to give statements in a unified way. For instance, the fact that the pullback of vector fields commutes with the Lie bracket reads as follows:

```
lemma pullback_lieBracket
  (hn : minSmoothness 𝕜 2 ≤ n) (h'f : ContDiffAt 𝕜 n f x)
  (hV : DifferentiableAt 𝕜 V (f x))
  (hW : DifferentiableAt 𝕜 W (f x)) :
  pullback 𝕜 f (lieBracket 𝕜 V W) x =
   lieBracket 𝕜 (pullback 𝕜 f V) (pullback 𝕜 f W) x
```

This statement is proved first without the `hn` assumption, requiring instead that the second derivative of f at x is symmetric to cancel out the terms (5.1). The theorem `ContDiffAt.isSymmSndFDerivAt` is then applied to show that the symmetry is satisfied under `hn`.

As an example of application, we can construct the Lie algebra of Lie groups over arbitrary fields. A Lie group G is a manifold with a multiplication and an inversion which are C^n for some n. Given two elements v and w of the tangent space $\mathfrak{g}$ at the identity, one can form the corresponding left-invariant vector fields V and W by transporting the vectors everywhere in the group by left-multiplication. The value at the identity of the Lie bracket $[V, W]$ of these vector fields is a new vector in $\mathfrak{g}$, called the (group) Lie bracket of v and w. This endows $\mathfrak{g}$ with a Lie algebra structure, as the Lie bracket of vector fields satisfies the required properties (linearity, antisymmetry, Jacobi identity). All these properties are checked by direct computations in vector spaces, and transported to the manifold since the pullback of the Lie bracket of vector fields under local diffeomorphisms is well defined as explained above. This requires some amount of smoothness, depending on the base field: over $\mathbb{R}$ or $\mathbb{C}$, we need C^3 for this construction, while over other fields C^ω is needed. The formalized version is as follows:

```
instance instLieAlgebraGroupLieAlgebra
  {I : ModelWithCorners 𝕜 E H} [TopologicalSpace G]
  [ChartedSpace H G] [Group G]
  [LieGroup I (minSmoothness 𝕜 3) G] [CompleteSpace E] :
  LieAlgebra 𝕜 (GroupLieAlgebra I G)
```

The details of the manifold or Lie group implementations can be safely ignored by the reader here. The main point in this instance is the assumption `LieGroup I (minSmoothness 𝕜 3) G` requiring that multiplication and inversion are C^3 if the field is $\mathbb{R}$ or $\mathbb{C}$ and analytic otherwise, in a unified way.

Remark 5.3. There are other approaches to the Lie algebra of a Lie group. For instance, it is straightforward to construct a Lie algebra structure on the space of left-invariant derivations, i.e., maps D from C^n functions to $\mathbb{R}$ verifying $D(fg) = f(1)D(g) + g(1)D(f))$: one sets $[D, D']f = D(D'f) - D'(Df)$, and it follows from the derivation property that this Lie bracket is again a derivation. In C^∞ finite-dimensional real manifolds, there is a canonical identification between the tangent space and the space of derivations on C^∞ functions. Therefore, in this specific case, this constructs the Lie algebra structure on the tangent space at the identity of a Lie group. This is the approach taken in [SF25]. However, it fails over other fields, or for smoothness classes other than C^∞, as the identification between the tangent space and the space of derivations fails in this generality. This approach is therefore not suitable for the generality level of `mathlib`.

6. Conclusion

We have described the formalization of the scale of smooth functions in the Lean mathematical library `mathlib`. It defines C^n functions for $n \in \mathbb{N} \cup \{\infty, \omega\}$, and departs in several ways from standard textbooks definition. The design is motivated by the applications, notably to various kinds of manifolds, that should be covered by the definition. The three main constraints are that we should allow C^n functions on domains, with respect to fields which can be more general than $\mathbb{R}$ or $\mathbb{C}$, with a hierarchy containing analytic functions. We have shown through several mathematical examples that naive definitions taking these constraints into account would fail natural properties that a good scale of C^n functions should have, forcing us to have a more elaborate definition.

The current definition is the result of a refactoring process that has been ongoing for several years: whenever an application could not be implemented, the definition had to be modified. The integration of analytic functions into the hierarchy of C^n functions, in particular, is motivated by the desire to be able to formalize Lie groups over arbitrary fields, with regularity at least C^3 over $\mathbb{R}$ or $\mathbb{C}$ and with analytic regularity otherwise. This application has been completely formalized, which attests to the effectiveness of the current definition, particularly in terms of providing unified formulations of definitions and results across the entire smoothness class.

We have not covered in this text the applications of C^n functions to analysis, although they are also very important. Let us just mention that the Schwartz space of superpolynomially decaying C^∞ functions is endowed in `mathlib` with its locally convex topological vector space structure by constructing seminorms involving the norms of the iterated derivatives at various points. Moreover, we have formalized the fact that the Fourier transform is an isomorphism of the Schwartz space, and the Fourier inversion formula. These results rely on many of the properties of C^n functions described in this paper.

References

[ACR18] Reynald Affeldt, Cyril Cohen, and Damien Rouhling, *Formalization techniques for asymptotic reasoning in classical analysis*, Journal of Formalized Reasoning **11** (2018), no. 1, 43–76. Cited page 2.

[Bou71] Nicolas Bourbaki, *éléments de mathématique. Fasc. XXXVI. Variétés différentielles et analytiques. Fascicule de résultats (Paragraphes 8 à 15)*, Actualités Scientifiques et Industrielles [Current Scientific and Industrial Topics], vol. No. 1347, Hermann, Paris, 1971. 281115. Cited pages 2, 11, and 19.

[Car71] Henri Cartan, *Differential calculus*, Hermann, Paris; Houghton Mifflin Co., Boston, MA, 1971, Exercises by C. Buttin, F. Rideau and J. L. Verley, Translated from the French. 344032. Cited pages 2 and 17.

[dMKA+15] Leonardo de Moura, Soonho Kong, Jeremy Avigad, Floris van Doorn, and Jakob von Raumer, *The Lean theorem prover (system description)*, Automated deduction—CADE 25, Lecture Notes in Comput. Sci., vol. 9195, Springer, Cham, 2015, pp. 378–388. 3446905. Cited page 1.

[HIH13] Johannes Hölzl, Fabian Immler, and Brian Huffman, *Type classes and filters for mathematical analysis in Isabelle/HOL*, Interactive theorem proving, Lecture Notes in Comput. Sci., vol. 7998, Springer, Heidelberg, 2013, pp. 279–294. 3111278. Cited pages 2 and 3.

[IZ19] Fabian Immler and Bohua Zhan, *Smooth manifolds and types to sets for linear algebra in Isabelle/HOL*, CPP 2019, Association for Computing Machinery, 2019, pp. 65–77. Cited page 3.

[mat20] mathlib community, *The Lean mathematical library*, Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2020, 2020, pp. 367–381. Cited page 1.

[Muj86] Jorge Mujica, *Complex analysis in Banach spaces*, North-Holland Mathematics Studies, vol. 120, North-Holland Publishing Co., Amsterdam, 1986. MR842435. Cited page 6.

[Pös21] Jürgen Pöschel, *On the Siegel-Sternberg linearization theorem*, J. Dynam. Differential Equations **33** (2021), no. 3, 1399–1425. 4299973. Cited page 8.

[Sch84] Wilhelmus H. Schikhof, *Ultrametric calculus*, Cambridge Studies in Advanced Mathematics, vol. 4, Cambridge University Press, Cambridge, 1984, An introduction to p-adic analysis. 791759. Cited pages 4 and 18.

[SF25] Richard Schmoetten and Jacques D. Fleuriot, *Constructing the Lie algebra of smooth vector fields on a Lie group in Isabelle/HOL*, J. Automat. Reason. **69** (2025), no. 3, Paper No. 18, 29. 4925296. Cited page 21.

IRMAR, CNRS UMR 6625, Université de Rennes, 35042 Rennes, France
Email address: `sebastien.gouezel@univ-rennes1.fr`