The derivative package

Written by:
Simon Jensen
sjelatex@gmail.com

The `derivative` package provides a set of commands which makes writing ordinary and partial derivatives of arbitrary order in a straight forward manner. Additionally, this package provides a set of commands to define variants of the aforementioned derivatives. A set of optional arguments along with lots of package options allow for easy and great flexibility over the derivative's format, such as where the function is positioned, point of evaluation, and switching between fraction styles. Moreover, the mixed order of the partial derivative and variants hereof is automatically computed. This package is written in the `expl3` language and requires therefore the LaTeX3 package bundles `l3kernel` and `l3package`. Additionally, the `mleftright` package is optional and provides the improved automatically scaling `\mleft` and `\mright`.

Contents

1 Introduction

This package originated as a personal package I developed several years ago for various projects. Initially written in TeX and LaTeX, it encountered numerous errors as its complexity grew, eventually becoming a convoluted code. Consequently, I rewrote the code using the LaTeX3 language, allowing for easier maintenance. Originally, I created this package due to the lack of robust derivative packages available. However, I later discovered the diffcoeff package, which served the purpose well. Unfortunately, by that time, I had already written a most of the code, undocumented. To address this, I decided to document the code and transform it into a publicly available package.

Regarding terminology, I use the abbreviation `inf` to represent the operator symbols d, ∂, δ, etc., commonly used in derivatives (e.g., $\frac{\mathrm{d}y}{\mathrm{d}x}, \frac{\partial y}{\partial x}, \frac{\delta y}{\delta x}$) and differentials (e.g., $\mathrm{d}x, \partial_x, \delta x$). In the description of macros and options, I often use the notation *cs-⟨placeholder⟩* to indicate a comma-separated list of ⟨*placeholder*⟩. For instance, [⟨*cs-numbers*⟩] is used in the option section to denote a comma-separated list of numbers for math space keys. It is important to note that whenever an argument specifies ⟨*keyvalue list*⟩, it refers to a comma-separated list of key-value pairs.

The GitHub repository for this package can be accessed at:
www.github.com/sjelatex/derivative.

2 Derivative

*[⟨*keyval list*⟩]{⟨*function*⟩}/!{⟨*variables*⟩}_{⟨*point₁*⟩}^{⟨*point₂*⟩}

The partial derivative is defined using the `\pdv` command, which has both mandatory and optional arguments. These arguments allow for the customization of specific parts and the style of the derivative.

```
\DeclareDerivative{\pdv}{\partial}[style-var=multiple, style-var-/=multiple,
style-var-!=mixed, style-var-/!=multiple, delims-eval=(), delims-eval-/=(),
delims-eval-!=()]
```

* The optional first argument of `\pdv` controls the placement of the function in relation to the fraction. By setting `switch-*=false`, the function is typeset in the numerator when the star is absent, and next to the fraction when the star is present. Here is an example:

$$\text{\texttt{\textbackslash pdv\{f\}\{x,y\}}} \quad \Longrightarrow \quad \frac{\partial^2 f}{\partial x\, \partial y}$$

$$\text{\texttt{\textbackslash pdv*\{f\}\{x,y\}}} \quad \Longrightarrow \quad \frac{\partial^2}{\partial x\, \partial y} f$$

The behavior of the star can be reversed by setting `switch-*=true`. In other words, the equations in the previous example will be interchanged.

The second argument of `\pdv`, enclosed in square brackets, is optional and used to specify options for the derivative using a `key=value` syntax. For instance, the order of differentiation can be set using the `order` option. Here is an example:

$$\text{\texttt{\textbackslash pdv[order=\{2,3\}]\{f\}\{x,y,z\}}} \quad \Longrightarrow \quad \frac{\partial^6 f}{\partial x^2\, \partial y^3\, \partial z}$$

$$\text{\texttt{\textbackslash pdv[order=\{\textbackslash beta,a,n+2a\}]\{f\}\{x,y,z\}}} \quad \Longrightarrow \quad \frac{\partial^{3a+\beta+n} f}{\partial x^\beta\, \partial y^a\, \partial z^{n+2a}}$$

$$\text{\texttt{\textbackslash pdv[order=\{2,n\^{}2,n\^{}2-1\}]\{f\}\{x,y,z\}}} \quad \Longrightarrow \quad \frac{\partial^{2n^2+1} f}{\partial x^2\, \partial y^{n^2}\, \partial z^{n^2-1}}$$

$$\text{\texttt{\textbackslash pdv[order=\{3/2-n/3,n/2,1/3\}]\{f\}\{x,y,z\}}} \quad \Longrightarrow \quad \frac{\partial^{n/6+11/6} f}{\partial x^{3/2-n/3}\, \partial y^{n/2}\, \partial z^{1/3}}$$

For a comprehensive list of available options that can be applied to derivatives, please refer to section 4.3.

{⟨*function*⟩} The first mandatory argument of `\pdv` is used to typeset the function that will be differentiated. Here are some examples:

$$\text{\texttt{\textbackslash pdv\{f(x,y,z)\}\{x,y,z\}}} \quad \Longrightarrow \quad \frac{\partial^3 f(x,y,z)}{\partial x\, \partial y\, \partial z}$$

$$\text{\texttt{\textbackslash pdv\{e\^{}x\textbackslash sin(y)\textbackslash ln(z)\}\{x,y,z\}}} \quad \Longrightarrow \quad \frac{\partial^3 e^x \sin(y)\ln(z)}{\partial x\, \partial y\, \partial z}$$

The function is displayed either in the numerator or to the right of the derivative, depending on the presence or absence of the star argument.

/

The fourth argument of `\pdv` is an optional slash that appears between the function and the variable arguments, indicating an alternative style. Its behaviour depends on the presence or absence of the exclamation mark argument.

When the exclamation mark is absent, the slash determines the fraction style in which the derivative is typeset. By default, when the slash is absent, the derivative is typeset with `\frac`, and when the slash is present, it is typeset with `\slashfrac`[1], as shown in the following example:

$$\text{\texttt{\textbackslash pdv\{f\}\{x,y\}}} \quad \Longrightarrow \quad \frac{\partial^2 f}{\partial x\,\partial y}$$

$$\text{\texttt{\textbackslash pdv\{f\}/\{x,y\}}} \quad \Longrightarrow \quad \partial^2 f \big/ \partial x\,\partial y$$

However, when the exclamation mark argument is present, the slash argument switches between shorthand styles rather than fraction styles.

Similar to the star argument, the effect of the slash's presence can be reversed by setting `switch-/=true`. In other words, the equations in the previous example will be interchanged.

!

The fifth argument of `\pdv` is an optional exclamation mark that appears between the function and the variable arguments. It allows switching the derivative into shorthand style, as described in section 4.3.

New: v1.3

When `switch-!=false` is used, along with the `\pdv`'s default shorthand styles, the derivative is typeset as shown in the example below:

$$\text{\texttt{\textbackslash pdv\{f\}!\{x,y\}}} \quad \Longrightarrow \quad \partial_x\,\partial_y f$$

$$\text{\texttt{\textbackslash pdv\{f\}/!\{x,y\}}} \quad \Longrightarrow \quad \partial x\,\partial y f$$

The effect of the exclamation mark's presence can be reversed by setting `switch-!=true`. In other words, the equations in the previous example will be interchanged.

Note. The order of the slash and exclamation mark is important. For example, `\pdv{f}!/{x,y}` will give $\partial_/f x, y$ and not the indented output $\partial x\,\partial y f$.

{⟨*variables*⟩}

This is the second and final mandatory argument is used to typeset the variable in which the function is differentiated with respect to. The variables should be provided as a comma-separated list

$$\text{\texttt{\textbackslash pdv\{f\}\{x\}}} \quad \Longrightarrow \quad \frac{\partial f}{\partial x}$$

$$\text{\texttt{\textbackslash pdv\{f\}\{x,y\}}} \quad \Longrightarrow \quad \frac{\partial^2 f}{\partial x\,\partial y}$$

---

[1]Note that `\slashfrac` is a macro defined by the package, please refer to section 7.1 for more details.

$_{\{\langle point_1\rangle\}}{}^{\{\langle point_2\rangle\}}$ The last optional argument specifies the point(s) of evaluation or variables held constant. It is an *e-type* argument from the xparse package, denoted as e{_^}. This means that the subscript _ and superscript ^ accepts an argument given within braces. The order of the subscript and superscript is independent, as shown in the following examples:

$$\texttt{\textbackslash pdv\{f\}\{x,y\}\_\{(x\_1,y\_1)\}} \quad \Longrightarrow \quad \left(\frac{\partial^2 f}{\partial x\, \partial y} \right)_{(x_1,y_1)}$$

$$\texttt{\textbackslash pdv\{f\}\{x,y\}\^{}\{(x\_2,y\_2)\}} \quad \Longrightarrow \quad \left(\frac{\partial^2 f}{\partial x\, \partial y} \right)^{(x_2,y_2)}$$

$$\texttt{\textbackslash pdv\{f\}\{x,y\}\_\{(x\_1,y\_1)\}\^{}\{(x\_2,y\_2)\}} \quad \Longrightarrow \quad \left(\frac{\partial^2 f}{\partial x\, \partial y} \right)^{(x_2,y_2)}_{(x_1,y_1)}$$

$$\texttt{\textbackslash pdv\{f\}\{x,y\}\^{}\{(x\_2,y\_2)\}\_\{(x\_1,y\_1)\}} \quad \Longrightarrow \quad \left(\frac{\partial^2 f}{\partial x\, \partial y} \right)^{(x_2,y_2)}_{(x_1,y_1)}$$

The subscript argument is commonly used to indicate the point of evaluation or the variables held constant. If needed, the superscript argument can be used to denote a second point of evaluation.

2.1 Other derivatives

In addition to the partial derivative, the package also provides five other derivative commands:

- Ordinary derivative: \odv - Used to represent the rate of change of a single-variable function with respect to its independent variable.

- Material derivative: \mdv - Applied in fluid mechanics and continuum mechanics to describe the rate of change of a quantity attached to a moving fluid particle.

- Functional derivative: \fdv - Used in functional analysis, calculus of variations, and quantum mechanics to express the derivative of a functional.

- Average rate of change: \adv - Used to denote the average rate of change of a quantity over a given interval.

- Jacobian: \jdv - Significant in multivariable calculus and linear algebra for representing the derivative of a vector-valued function.

A unique feature of this package is that it allows you to define your own derivatives, as described in section 5.1. This means you can create custom derivative commands tailored to your specific needs.

If you require more information on the usage and customization of these derivative commands, please refer to the documentation in section 5.1.

\odv *[⟨*keyval list*⟩]{⟨*function*⟩}/{⟨*variables*⟩}_{⟨*point₁*⟩}^{⟨*point₂*⟩}

In this package, the ordinary derivative is defined with an upright lowercase d if the package option `upright=true` is used. Otherwise, it is defined with an italic lowercase *d*. This choice was made to align with the convention used in many modern books.

```
\DeclareDerivative{\odv}{\mathrm{d}}                          upright=true
\DeclareDerivative{\odv}{\mathnormal{d}}                      italic=true
```

The ordinary derivative can then be typeset as:

$$\frac{\mathrm{d}f}{\mathrm{d}x} = \lim_{h \to 0} \left(\frac{f(x+h) - f(x)}{h} \right)$$

By adjusting the `style-inf` or `style-inf-num` and `style-inf-den` keys, the operator d can be customized to personal preference or specific formatting requirements.

\mdv *[⟨*keyval list*⟩]{⟨*function*⟩}/{⟨*variables*⟩}_{⟨*point₁*⟩}^{⟨*point₂*⟩}

The material derivative is defined with an upright uppercase D when the package option `upright=true`. Otherwise, it is defined with an italic uppercase D.

```
\DeclareDerivative{\mdv}{\mathrm{D}}                          upright=true
\DeclareDerivative{\mdv}{\mathnormal{D}}                      italic=true
```

In physics, the material derivative is defined by

$$\frac{\mathrm{D}\varphi(\mathbf{r}, t)}{\mathrm{D}t} := \frac{\partial \varphi(\mathbf{r}, t)}{\partial t} + \dot{\mathbf{r}} \cdot \nabla \varphi(\mathbf{r}, t)$$

By adjusting the `style-inf` or `style-inf-num` and `style-inf-den` keys, the operator D can be customized to personal preference or specific formatting requirements.

\fdv *[⟨*keyval list*⟩]{⟨*function*⟩}/{⟨*variables*⟩}_{⟨*point₁*⟩}^{⟨*point₂*⟩}

In this package, the functional derivative is defined with a lowercase delta. By default, it is represented in italic form. The functional derivative can be defined as follows:

```
\DeclareDerivative{\fdv}{\delta}
```

In physics, the functional derivative is commonly employed in various equations, such as the Lagrange equation and the derivation of the Hartree-Fock equation. Examples of its usage include:

$$\frac{\delta I}{\delta q_\alpha} = \frac{\partial L}{\partial q_\alpha} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}_\alpha} = 0, \qquad \frac{\delta \mathcal{L}}{\delta \psi_n^*} = \hat{F}|\psi_n\rangle - \epsilon_n |\psi_n\rangle = 0,$$

\adv *[⟨*keyval list*⟩]{⟨*function*⟩}/{⟨*variables*⟩}_{⟨*point₁*⟩}^{⟨*point₂*⟩}

The average rate of change is defined to use an upright uppercase delta with the default settings. In this package, the average rate of change is defined as

```
\DeclareDerivative{\adv}{\Delta}
```

The average rate of change can be expressed as

$$\frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1}$$

\jdv   *[⟨*keyval list*⟩]{⟨*function*⟩}/{⟨*variables*⟩}_{⟨*point₁*⟩}^{⟨*point₂*⟩}

In this package, the Jacobian is defined with an italic partial differential by default. Additionally, a pair of parentheses is automatically inserted around the function and variable.

```
\DeclareDerivative{\jdv}{\partial}[fun=true, var=1]
```

which gives

$$\frac{\partial(f, g, h)}{\partial(x, y, z)}$$

3 Differentials

\odif $*[\langle keyval\ list\rangle]\{\langle variables\rangle\}$

The differential `\odif` is defined with both mandatory and optional arguments that allow for customizing its typesetting and style. It is defined with an upright lowercase d when `upright=true` is used. Otherwise, it will be defined with an italic lowercase d.

```
\DeclareDifferential{\odif}{\mathrm{d}}          upright=true
\DeclareDifferential{\odif}{\mathnormal{d}}      italic=true
```

* The first argument of `\odif` is an optional star that determines the notation style of the differential. With the `switch-*=false` option, the variables and orders are typeset as subscript and superscript, respectively, when the star is present. When the star is absent, the infinitesimal is placed in front of each variable, as illustrated below:

$$\texttt{\textbackslash odif\{x,y,z\}} \quad \implies \quad \mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z$$

$$\texttt{\textbackslash odif*\{x,y,z\}} \quad \implies \quad \mathrm{d}_{x,y,z}$$

The behavior of the star can be reversed by setting `switch-*=true`. In other words, the equations in the previous example will be interchanged.

$[\langle keyval\ list\rangle]$ The second argument, enclosed in square brackets, is optional and is used to specify options for the differential using the `key=value` syntax. Here are some examples:

$$\texttt{\textbackslash odif[order=\{n,3\}]\{x,y,z\}} \quad \implies \quad \mathrm{d}^n x\,\mathrm{d}^3 y\,\mathrm{d}z$$

$$\texttt{\textbackslash odif[sep-var-inf=0]\{x,y,z\}} \quad \implies \quad \mathrm{d}x\mathrm{d}y\mathrm{d}z$$

$$\texttt{\textbackslash odif*[sep-var-var=0]\{x,y,z\}} \quad \implies \quad \mathrm{d}_{xyz}$$

The available options for the differential can be found in section 4.4

$\{\langle variables\rangle\}$ The mandatory argument is used to typeset the variables in the differential. Here are some examples:

$$\texttt{\textbackslash odif\{x\}} \quad \implies \quad \mathrm{d}x$$

$$\texttt{\textbackslash odif\{s\_1,s\_2...,s\_n\}} \quad \implies \quad \mathrm{d}s_1\,\mathrm{d}s_2...\,\mathrm{d}s_n$$

3.1 Other differentials

In addition to the regular differential, the package also provides four other differential commands:

- Partial differential: `\pdif` - This command is used to typeset partial differentials.

- Uppercase D: `\mdif` - It is used to typeset differentials with an uppercase "D".

- Lowercase delta: `\fdif` - This command is used to typeset differentials with a lowercase delta symbol.

9

- Uppercase delta: `\adif` - It is used to typeset differentials with an uppercase Delta symbol.

A unique feature of this package is that it allows you to define your own differentials, as described in section 5.2. This means you can create custom differential commands tailored to your specific needs.

If you require more information on the usage and customization of these differential commands, please refer to the documentation in section 5.2.

`\pdif` *[⟨*keyval list*⟩]{⟨*variables*⟩}*

The partial differential `\pdif` is commonly used as a shorthand notation for the partial derivative. In this package, it is defined as follows:

```
\DeclareDifferential{\pdif}{\partial}[style-notation=single,
style-notation-*=mixed]
```

The non-star and star versions are represented on the left side as:

$$\partial_x^i \, \partial_y^j \, \partial_z^k := \frac{\partial^{i+j+k}}{\partial x^i \, \partial y^j \, \partial z^k}$$

$$\partial^i x \, \partial^j y \, \partial^k z := \frac{\partial^{i+j+k}}{\partial x^i \, \partial y^j \, \partial z^k}$$

respectively.

`\mdif` *[⟨*keyval list*⟩]{⟨*variables*⟩}*

Another commonly used shorthand notation for derivatives is the differential with an uppercase D. In this package it is defined with a upright D when `upright=true`. Otherwise, it is defined with an italic D.

```
\DeclareDifferential{\mdif}{\mathrm{D}}[style-notation=single,
style-notation-*=mixed]
\DeclareDifferential{\mdif}{\mathnormal{D}}[style-notation=single,
style-notation-*=mixed
```

`upright=true`

`italic=true`

The non-star and star version gives

$$\mathrm{D}_x^i \, \mathrm{D}_y^j \, \mathrm{D}_z^k$$
$$\mathrm{D}^i x \, \mathrm{D}^j y \, \mathrm{D}^k z$$

respectively.

`\fdif` *[⟨*keyval list*⟩]{⟨*variables*⟩}*

When working with functional derivatives, another commonly used differential is one that uses a delta symbol. It is defined as follows:

```
\DeclareDifferential{\fdif}{\delta}
```

For example, in expression like:

$$\delta J = \int_a^b \frac{\partial L}{\partial f} \delta f(x) + \frac{\partial L}{\partial f'} \frac{\mathrm{d}}{\mathrm{d}x} \delta f(x) \, \mathrm{d}x$$

this differential is frequently encountered.

\adif   *[⟨*keyval list*⟩]{⟨*variables*⟩}

A differential for differences that uses a uppercase delta is defined as

```
\DeclareDifferential{\adif}{\Delta}
```

For example, the difference between two energy levels can be written as:

$$\Delta E = E_2 - E_1$$

4 Options

This package accepts its options using the familiar *key=value* syntax. The keys are divided into categories, with each key having its associated category as a prefix.

`\derivset`   `{⟨command⟩}[⟨keyval list⟩]`

Updated: v1.0

The `\derivset` command is used to set default values for derivative and differential options in the preamble. While it can be used within the document, the new `[⟨keyval list⟩]` arguments in the derivative and differential commands allow for more flexibility in specifying options on a per-use basis.

`{⟨command⟩}`   The mandatory argument determines which command the *key=value* pairs are assigned to. The allowed ⟨*commands*⟩ are the derivatives and differentials defined by the package, as well as any additional derivatives and differentials you have defined (see sections 5.1 and 5.2 for more information). Additionally, the special value `all` is allowed, which provides access to the options that apply to all derivatives and differentials.

$$\texttt{\textbackslash derivset\{\textbackslash odv\}[switch-*=true] \textbackslash odv\{y\}\{x\}} \quad \Longrightarrow \quad \frac{\mathrm{d}}{\mathrm{d}x}y$$

$$\texttt{\textbackslash derivset\{\textbackslash odif\}[switch-*=true] \textbackslash odif\{y\}\{x\}} \quad \Longrightarrow \quad \mathrm{d}_y x$$

`[⟨keyval list⟩]`   The optional argument accepts input as a comma-separated list of *key=value* pairs. If `[⟨keyval list⟩]` is omitted, the options will be set to the package's default settings for the chosen `{⟨command⟩}`. For example, using the `\derivset{\odv}` command will set the options for the ordinary derivative to the default settings defined by the package.

$$\texttt{\textbackslash derivset\{\textbackslash odv\}[switch-*=false] \textbackslash odv\{y\}\{x\} \textbackslash odv*\{y\}\{x\}} \quad \Longrightarrow \quad \frac{\mathrm{d}y}{\mathrm{d}x}\frac{\mathrm{d}}{\mathrm{d}x}y$$

$$\texttt{\textbackslash derivset\{\textbackslash odv\}[switch-*=true] \textbackslash odv\{y\}\{x\} \textbackslash odv*\{y\}\{x\}} \quad \Longrightarrow \quad \frac{\mathrm{d}}{\mathrm{d}x}y\frac{\mathrm{d}y}{\mathrm{d}x}$$

$$\texttt{\textbackslash derivset\{\textbackslash odv\} \textbackslash odv\{y\}\{x\} \textbackslash odv*\{y\}\{x\}} \quad \Longrightarrow \quad \frac{\mathrm{d}y}{\mathrm{d}x}\frac{\mathrm{d}}{\mathrm{d}x}y$$

4.1 Categories

This section seeks to give a detailed description of each category.

- The `style-⟨…⟩` keys allow you to customize the style of derivatives or differentials by specifying options such as the fraction command, infinitesimal notation (e.g., d, ∂) and its font, and variable treatment.

- The `delims-⟨…⟩` keys sets the delimiters used around the ⟨…⟩. The Rule of Two applies: 'Always two there are, no more, no less. A left and a right delimiter'. Only delimiters that can be scaled with commands like `\left`, `\big`, etc., are allowed.

- The `scale-`⟨…⟩ keys sets the size of the ⟨…⟩'s delimiters. The values `big`, `Big`, `bigg`, and `Bigg` are self-explanatory. The value `none` leaves the delimiters unscaled (except for periods, which are removed). The value `auto` automatically scales the delimiters using `\left` and `\right`.

- The `sep-`⟨…⟩`-`⟨…⟩ keys inserts their value between ⟨…⟩ and ⟨…⟩. If the value is a comma-separated list of up to three numbers (e.g. `{1,2,3}`) it is converted into the syntax `\muskip 1 mu plus 2 mu minus 3 mu` and used accordingly. For other values, they are used as given without any conversion.

- The `switch-`⟨…⟩ keys allow you to change the behaviour of an argument by swapping the effect of the presence or absence of an optional character argument.

- The `sort-`⟨…⟩ keys handle the sorting algorithm used for the mixed order. These keys allow you to choose the sorting method that best suits your preferences.

- The miscellaneous keys do not fall into any of the previously mentioned categories and does not have a prefix.

Note. A value with a superscripted `R` indicates that it requires a specific package to be loaded. Additionally, some keys have multiple versions, denoted by `-/`, `-!`, and `-/!` at the end. These keys are specifically related to the slash and exclamation mark arguments.

4.2 Package options

The package options can be used with the following syntax when loading the package in the preamble:

$$\texttt{\textbackslash usepackage[}\langle \textit{keyval list}\rangle \texttt{]\{derivative\}}$$

| italic | true, false | false |

New: v1.1 — Sets the font style of the infinitesimals d and D used in `\odv`, `\mdv`, `\odif` and `\mdif` to italic using `\mathnormal`. The `italic` and `upright` options are mutually exclusive.

| upright | true, false | true |

New: v1.1 — Sets the font style of the infinitesimals d and D used in `\odv`, `\mdv`, `\odif` and `\mdif` to upright using `\mathrm`. The `italic` and `upright` options are mutually exclusive.

4.3 Derivative options

The options in this subsection are available fo customizing the behaviour of derivatives defined by the package and you.

Style

<div>

style-inf

⟨*math-font-style*⟩⟨*infinitesimal*⟩

Sets the infinitesimal used in the derivative. This is a meta key, which means it sets the value of both `style-inf-num` and `style-inf-den` at the same time.

$$\text{\odv[style-inf=\symbf{d}]{f}{x}} \implies \frac{\mathbf{d}f}{\mathbf{d}x}$$

style-inf-num

⟨*math-font-style*⟩⟨*infinitesimal*⟩   d

Sets the infinitesimal used in the numerator of the derivative. This option is also used in the shorthand versions, i.e., when the exclamation mark argument is used. The default infinitesimal is a plain d.

$$\text{\odv[style-inf-num=\mathbf{d}]{f}{x}} \implies \frac{\mathbf{d}f}{\mathrm{d}x}$$

$$\text{\odv[style-inf-num=\mathbf{d}]{f}!{x}} \implies \mathbf{d}xf$$

style-inf-den

⟨*math-font-style*⟩⟨*infinitesimal*⟩   d

Sets the infinitesimal used in the denominator of the derivative. The default infinitesimal is a plain d.

$$\text{\odv[style-inf-den=\mathbf{d}]{f}{x}} \implies \frac{\mathrm{d}f}{\mathbf{d}x}$$

style-frac

⟨*fraction*⟩   \frac

The derivative uses the fraction set by this key when the slash argument is absent. If `switch-/=true`, the derivative uses this fraction when the slash argument is present. The key's default value is the usual fraction `\frac`.

$$\text{\odv[style-frac=\tfrac]{f}{x}} \implies \tfrac{\mathrm{d}f}{\mathrm{d}x}$$

$$\text{\odv[switch-/=true, style-frac=\tfrac]{f}/{x}} \implies \tfrac{\mathrm{d}f}{\mathrm{d}x}$$

style-frac-/

⟨*fraction*⟩   \slashfrac

The derivative uses the fraction set by this key when the slash argument is present. If `switch-/=true`, the derivative uses this fraction when the slash argument is absent. The key's default value is a text-styled fraction `\slashfrac`[1] on the form $\mathrm{d}f/\mathrm{d}x$.

$$\text{\odv[style-frac-/=\sfrac]{f}/{x}} \implies {}^{\mathrm{d}f}/_{\mathrm{d}x}$$

style-var

single, multiple   single

This option determines how the derivative treats its variables and orders when both the slash argument and the exclamation mark are absent assuming `switch-/=false` and `switch-!=false`

single The derivative behaves like a ordinary derivative, treating the variable argument and order as single entities.

</div>

multiple The derivative behaves like a partial derivative, treating the variable argument as a list of comma-separated variables. The mixed order is automatically calculated and sorted based on the `order=⟨cs-orders⟩` key. See section 6 for limitations on the automatic calculation.

$$\pdv[style-var=single, order=\{1,2,3\}]\{f\}\{x,y,z\} \quad \implies \quad \frac{\partial^{1,2,3} f}{\partial x, y, z^{1,2,3}}$$

$$\pdv[style-var=multiple]\{f\}\{x,y,z\} \quad \implies \quad \frac{\partial^3 f}{\partial x\, \partial y\, \partial z}$$

style-var-/

New: v1.3

single, multiple   single

This option behaves identically to `style-var`, however it is used when the slash argument is present and the exclamation mark is absent assuming `switch-/=false` and `switch-!=false`.

$$\pdv[style-var-/=single, order=\{1,2,3\}]\{f\}/\{x,y,z\} \quad \implies \quad \partial^{1,2,3} f \big/ \partial x, y, z^{1,2,3}$$

$$\pdv[style-var-/=multiple]\{f\}/\{x,y,z\} \quad \implies \quad \partial^3 f \big/ \partial x\, \partial y\, \partial z$$

style-var-!

New: v1.3

single, multiple, mixed   multiple

This options determines the shorthand notation of the derivative when the slash argument is absent and the exclamation mark is present assuming `switch-/=false` and `switch-!=false`.

single Creates a single differential with variables as subscripts and orders as superscripts.

multiple Creates a separate differential for each variable.

mixed Creates a separate differential for each variable but with variables as subscripts and orders as superscripts.

$$\pdv[style-var-!=single]\{f\}!\{x,y,z\} \quad \implies \quad \partial_{x,y,z} f$$

$$\pdv[style-var-!=multiple]\{f\}!\{x,y,z\} \quad \implies \quad \partial x\, \partial y\, \partial z f$$

$$\pdv[style-var-!=mixed]\{f\}!\{x,y,z\} \quad \implies \quad \partial_x\, \partial_y\, \partial_z f$$

style-var-/!

New: v1.3

single, multiple, mixed   single

This option behaves identical to `style-var-!`, however it is used when both the exclamation mark and slash arguments are present assuming `switch-/=false` and `switch-!=false`.

$$\pdv[style-var-/!=single]\{f\}/!\{x,y,z\} \quad \implies \quad \partial_{x,y,z} f$$

$$\pdv[style-var-/!=multiple]\{f\}/!\{x,y,z\} \quad \implies \quad \partial x\, \partial y\, \partial z f$$

$$\pdv[style-var-/!=mixed]\{f\}/!\{x,y,z\} \quad \implies \quad \partial_x\, \partial_y\, \partial_z f$$

Scaling

scale-eval `auto, none, big, Big, bigg, Bigg`   `auto`

Sets the size of the delimiters used for the point of evaluation. It is applied when the slash argument is absent, assuming `switch-/=false`. The default behaviour is to automatically scale the delimiters.

$$\texttt{\textbackslash pdv[scale-eval=auto]\{f\}\{x,y\}\_\{(x\_0,y\_0)\}} \implies \left(\frac{\partial^2 f}{\partial x\,\partial y}\right)_{(x_0,y_0)}$$

$$\texttt{\textbackslash pdv[scale-eval=auto]\{f\}!\{x,y\}\_\{(x\_0,y\_0)\}} \implies \left(\partial_x\,\partial_y f\right)_{(x_0,y_0)}$$

scale-eval-/ `auto, none, big, Big, bigg, Bigg`   `auto`

Sets the size of the delimiters used for the point of evaluation. It is applied when the slash argument is present, assuming `switch-/=false`. The default behaviour is to automatically scale the delimiters.

$$\texttt{\textbackslash pdv[scale-eval-/=none]\{f\}/\{x,y\}\_\{(x\_0,y\_0)\}} \implies (\partial^2 f/\partial x\,\partial y)_{(x_0,y_0)}$$

scale-eval-! `auto, none, big, Big, bigg, Bigg`   `auto`

New: v1.3

Sets the size of the delimiters used for the point of evaluation. It is applied when the exclamation mark argument is present, assuming `switch-!=false`. The default behaviour is to automatically scale the delimiters. This option takes priority over `scale-eval-/` when the slash argument is also present.

$$\texttt{\textbackslash pdv[scale-eval-!=Big]\{f\}!\{x,y\}\_\{(x\_0,y\_0)\}} \implies \left(\partial_x\,\partial_y f\right)_{(x_0,y_0)}$$

scale-fun `auto, none, big, Big, bigg, Bigg`   `auto`

Sets the size of the delimiters used around the function. It is applied when `fun=true`. The default behaviour is to automatically scale the delimiters.

$$\texttt{\textbackslash pdv[scale-fun=big, fun]\{f\}\{x,y\}} \implies \frac{\partial^2 (f)}{\partial x\,\partial y}$$

scale-var `auto, none, big, Big, bigg, Bigg`   `auto`

Sets the size of the delimiters used around the variables specified by the `var={…}` option when the exclamation mark argument is absent assuming `switch-!=false`. The default behaviour is to automatically scale the delimiters.

$$\texttt{\textbackslash pdv[scale-var=Big, var]\{f\}\{x,y\}} \implies \frac{\partial^2 f}{\partial\Big(x\Big)\partial\Big(y\Big)}$$

scale-var-! `auto, none, big, Big, bigg, Bigg`   `auto`

New: v1.3

Sets the size of the delimiters used around the variables specified by the `var={…}` option when the exclamation mark argument is present assuming `switch-!=false`. The default behaviour is to automatically scale the delimiters.

$$\texttt{\textbackslash pdv[scale-var-!=big, var]\{f\}!\{x,y\}} \implies \partial_{(x)}\,\partial_{(y)}f$$

scale-frac	auto, none, big, Big, bigg, Bigg	auto

Sets the size of the delimiters used for around the fraction. It is applied when the slash argument is absent and `frac=true`, assuming `switch-/=false`. The default behaviour is to automatically scale the delimiters.

$$\texttt{\textbackslash pdv[scale-frac=bigg, frac]\{f\}\{x,y\}} \implies \left(\frac{\partial^2 f}{\partial x\,\partial y}\right)$$

scale-frac-/	auto, none, big, Big, bigg, Bigg	auto

Sets the size of the delimiters used for around the fraction. It is applied when the slash argument is present and `frac=true`, assuming `switch-/=false`. The default behaviour is to automatically scale the delimiters.

$$\texttt{\textbackslash pdv[scale-frac-/=Bigg, frac]\{f\}/\{x,y\}} \implies \left(\partial^2 f \big/ \partial x\,\partial y\right)$$

Delimiters

delims-eval	⟨*left delimiter*⟩⟨*right delimiter*⟩	. \rvert

Sets the delimiters used to indicate the point of evaluation. These delimiters are used when the slash argument is absent, assuming `switch-/=false`. The key's default delimiters are a period and a vertical line.

$$\texttt{\textbackslash pdv[delims-eval=.|]\{f\}\{x,y\}\_\{(x\_0,y\_0)\}} \implies \left.\frac{\partial^2 f}{\partial x\,\partial y}\right|_{(x_0,y_0)}$$

delims-eval-/	⟨*left delimiter*⟩⟨*right delimiter*⟩	. \rvert

Sets the delimiters used to indicate the point of evaluation. These delimiters are used when the slash argument is present, assuming `switch-/=false`. The key's default delimiters are a period and a vertical line.

$$\texttt{\textbackslash pdv[delims-eval-/=[]]\{f\}/\{x,y\}\_\{(x\_0,y\_0)\}} \implies \left[\partial^2 f\big/\partial x\,\partial y\right]_{(x_0,y_0)}$$

delims-eval-!	⟨*left delimiter*⟩⟨*right delimiter*⟩	. \rvert

Sets the delimiters used to indicate the point of evaluation. These delimiters are used when the slash argument is present, assuming `switch-!=false`. The key's default delimiters are a period and a vertical line. This option takes priority over `delims-eval-/` when the slash argument is also present.

$$\texttt{\textbackslash pdv[delims-eval-!=[]]\{f\}!\{x,y\}\_\{(x\_0,y\_0)\}} \implies \left[\partial_x\,\partial_y f\right]_{(x_0,y_0)}$$

delims-fun	⟨*left delimiter*⟩⟨*right delimiter*⟩	( )

Sets the delimiters used around the function. These delimiters are used when `fun=true`. The key's default delimiters are parentheses.

$$\verb|\pdv[delims-fun=\langle\rangle, fun]{f}{x,y}| \implies \frac{\partial^2 \langle f \rangle}{\partial x \, \partial y}$$

`delims-var`	⟨*left delimiter*⟩⟨*right delimiter*⟩	( )

Sets the delimiters used around the variables specified by the `var={…}` option when the exclamation mark argument is absent assuming `switch-!=false`. The key's default delimiters are parentheses.

$$\verb|\pdv[delims-var=\{\}, var]{f}{x,y}| \implies \frac{\partial^2 f}{\partial \{x\} \, \partial \{y\}}$$

`delims-var-!`	⟨*left delimiter*⟩⟨*right delimiter*⟩	( )

New: v1.3

Sets the delimiters used around the variables specified by the `var={…}` option when the exclamation mark argument is present assuming `switch-!=false`. The key's default delimiters are parentheses.

$$\verb|\pdv[delims-var-!=\lceil\rceil, var]{f}!{x,y}| \implies \partial_{\lceil x \rceil} \, \partial_{\lceil y \rceil} f$$

`delims-frac`	⟨*left delimiter*⟩⟨*right delimiter*⟩	( )

Sets delimiters used around the fraction in the derivative. These delimiters are used when the slash argument is absent and `frac=true`, assuming `switch-/=false`. The key's default delimiters are parentheses.

$$\verb|\pdv[delims-frac=||, frac]{f}{x,y}| \implies \left| \frac{\partial^2 f}{\partial x \, \partial y} \right|$$

$$\verb|\pdv*[delims-frac=||, frac=true]{y}{x}| \implies \left| \frac{\partial}{\partial x} \right| y$$

`delims-frac-/`	⟨*left delimiter*⟩⟨*right delimiter*⟩	( )

Sets delimiters used around the fraction in the derivative. These delimiters are used when the slash argument is present and `frac=true`, assuming `switch-/=false`. The key's default delimiters are parentheses.

$$\verb|\pdv[delims-frac-/=\|\|, frac]{f}/{x,y}| \implies \left\| \partial^2 f / \partial x \, \partial y \right\|$$

$$\verb|\pdv*[delims-frac-/=\|\|, frac]{f}/{x,y}| \implies \left\| \partial^2 / \partial x \, \partial y \right\| f$$

Math spacing and separators

`sep-inf-ord`	⟨*cs-number*⟩, {⟨*mspace*⟩}, {⟨*delimiter*⟩}	0

Sets the separator that is inserted to the left of the infinitesimal's power when the (mixed) order *is different* from 1. The default value corresponds to `0 mu`.

$$\verb|\pdv[sep-inf-ord=\here]{f}{x,y}| \implies \frac{\partial^{\langle \text{here} \rangle 2} f}{\partial x \, \partial y}$$

$$\verb|\pdv[sep-inf-ord=\here, order=2]{f}!{x,y}| \implies \partial_x^{\langle \text{here} \rangle 2} \, \partial_y f$$

$$\verb|\pdv[sep-inf-ord=\here, order=2]{f}/!{x,y}| \implies \partial^{\langle \text{here} \rangle 2} x \, \partial y f$$

sep-inf-fun $\langle$*cs-number*$\rangle$, {$\langle$*mspace*$\rangle$}, {$\langle$*delimiter*$\rangle$}   0

Sets the separator that is inserted between the infinitesimal and the function when the (mixed) order *is equal* to 1. The space is only inserted when a non-blank function is printed *in the numerator*. The default value corresponds to 0 mu.

$$\texttt{\textbackslash pdv[sep-inf-fun=\textbackslash here]\{f\}\{x\}} \quad \implies \quad \frac{\partial_{\langle\texttt{here}\rangle} f}{\partial x}$$

sep-ord-fun $\langle$*cs-number*$\rangle$, {$\langle$*mspace*$\rangle$}, {$\langle$*delimiter*$\rangle$}   0

Sets the separator that is inserted between the function and the infinitesimal's order when the order *is different* from 1. The space is only inserted when a non-blank function is printed *in the numerator*. The default value corresponds to 0 mu.

$$\texttt{\textbackslash pdv[sep-ord-fun=\textbackslash here]\{f\}\{x,y\}} \quad \implies \quad \frac{\partial^2{}_{\langle\texttt{here}\rangle} f}{\partial x \,\partial y}$$

sep-frac-fun $\langle$*cs-number*$\rangle$, {$\langle$*mspace*$\rangle$}, {$\langle$*delimiter*$\rangle$}   0

New: v1.0

Sets the separator that is inserted between the fractional part of the derivative and the function when a non-blank function is printed *next to the derivative*. The default value corresponds to 0 mu.

$$\texttt{\textbackslash pdv*[sep-frac-fun=\textbackslash here]\{f\}\{x,y\}} \quad \implies \quad \frac{\partial^2}{\partial x \,\partial y}{}_{\langle\texttt{here}\rangle} f$$

sep-inf-var $\langle$*cs-number*$\rangle$, {$\langle$*mspace*$\rangle$}, {$\langle$*delimiter*$\rangle$}   0

Sets the separator that is inserted between the infinitesimal and the following non-blank variable. In shorthand mode, the order must also *be different* from 1. The default value corresponds to 0 mu.

$$\texttt{\textbackslash pdv[sep-inf-var=\textbackslash here, order=2]\{f\}\{x,y\}} \quad \implies \quad \frac{\partial^3 f}{\partial_{\langle\texttt{here}\rangle} x^2 \,\partial_{\langle\texttt{here}\rangle} y}$$

$$\texttt{\textbackslash pdv[sep-inf-var=\textbackslash here, order=2]\{f\}!\{x,y\}} \quad \implies \quad \partial^2_{\langle\texttt{here}\rangle x} \,\partial_{\langle\texttt{here}\rangle y} f$$

$$\texttt{\textbackslash pdv[sep-inf-var=\textbackslash here, order=2]\{f\}/!\{x,y\}} \quad \implies \quad \partial^2 x \,\partial_{\langle\texttt{here}\rangle} y f$$

sep-var-ord $\langle$*cs-number*$\rangle$, {$\langle$*mspace*$\rangle$}, {$\langle$*delimiter*$\rangle$}   0

Sets the separator that is inserted to the left of the variable's power when its order *is different* from 1. The default value corresponds to 0 mu.

$$\texttt{\textbackslash pdv[sep-var-ord=\textbackslash here, order=n]\{f\}\{x,y\}} \quad \implies \quad \frac{\partial^{n+1} f}{\partial x^{\langle\texttt{here}\rangle n} \,\partial y}$$

sep-var-inf $\langle$*cs-number*$\rangle$, {$\langle$*mspace*$\rangle$}, {$\langle$*delimiter*$\rangle$}   \mathop{}\!

Sets the separator that is inserted between a variable and the following infinitesimal when more than one *non-blank* variables are given. In fraction mode, it is also required that the variable's order *is equal* to 1. The default value is \mathop{}\!.

$$\verb|\pdv[sep-var-inf=\here, order=n]{f}{x,y,z}| \implies \frac{\partial^{n+2} f}{\partial x^n \, \partial y_{\langle here\rangle}\partial z}$$

$$\verb|\pdv[sep-var-inf=\here, order=n]{f}!{x,y,z}| \implies \partial_x^n {}_{\langle here\rangle}\partial_y {}_{\langle here\rangle}\partial_z f$$

$$\verb|\pdv[sep-var-inf=\here, order=n]{f}/!{x,y,z}| \implies \partial^n x_{\langle here\rangle}\partial y_{\langle here\rangle}\partial z f$$

sep-ord-inf $\langle\textit{cs-number}\rangle$, {$\langle\textit{mspace}\rangle$}, {$\langle\textit{delimiter}\rangle$}   `\mathop{}\!`

Sets the separator that is inserted between an order and the following infinitesimal when more than one *non-blank* variables are given. In fraction mode, it is also required that the variable's order *is different* from 1. The default value is `\mathop{}\!`.

$$\verb|\pdv[sep-ord-inf=\here, order=n]{f}{x,y,z}| \implies \frac{\partial^{n+2} f}{\partial x^n {}_{\langle here\rangle}\partial y \, \partial z}$$

sep-ord-ord $\langle\textit{cs-number}\rangle$, {$\langle\textit{mspace}\rangle$}, {$\langle\textit{delimiter}\rangle$}   ,
New: v1.3

Sets the separator that is inserted between two orders when more than one *non-blank* variables are given in shorthand notation when it is set to `single`. The default value is a comma.

$$\verb|\odv[sep-ord-ord=\here, order={n,2}]{f}/!{x,y}| \implies \mathrm{d}^{n_{\langle here\rangle}2}_{x,y} f$$

sep-ord-var $\langle\textit{cs-number}\rangle$, {$\langle\textit{mspace}\rangle$}, {$\langle\textit{delimiter}\rangle$}   0
New: v1.3

Sets the separator that is inserted between an order and the following variable in shorthand notation when it is set to `multiple`. The default value corresponds to `0 mu`.

$$\verb|\odv[sep-ord-var=\here, order=n]{f}!{x,y}| \implies \mathrm{d}^n {}_{\langle here\rangle}x \, \mathrm{d}y f$$

sep-var-var $\langle\textit{cs-number}\rangle$, {$\langle\textit{mspace}\rangle$}, {$\langle\textit{delimiter}\rangle$}   ,
New: v1.0

Sets the separator that is inserted between two variables when more than one *non-blank* variables are given in shorthand notation when it is set to `single`. The default value is a comma.

$$\verb|\odv[sep-var-var=\here]{f}{x,y}| \implies \frac{\mathrm{d}f}{\mathrm{d}x_{\langle here\rangle}y}$$

$$\verb|\odv[sep-var-var=\here]{f}/!{x,y}| \implies \mathrm{d}_{x_{\langle here\rangle}y} f$$

sep-eval-sb $\langle\textit{cs-number}\rangle$, {$\langle\textit{mspace}\rangle$}, {$\langle\textit{delimiter}\rangle$}   0

Sets the separator that is inserted in the evaluation subscript left to the point of evaluation when a *non-blank* subscript is given. The default value corresponds to `0 mu`.

$$\verb|\pdv[sep-eval-sb=\here]{f}{x,y}_{(x_1,y_1)}| \implies \left(\frac{\partial^2 f}{\partial x \, \partial y}\right)_{\langle here\rangle(x_1,y_1)}$$

sep-eval-sp $\langle\textit{cs-number}\rangle$, {$\langle\textit{mspace}\rangle$}, {$\langle\textit{delimiter}\rangle$}   0

Sets the separator that is inserted in the evaluation superscript left to the point of evaluation when a *non-blank* superscript is given. The default value corresponds to `0 mu`.

$$\texttt{\textbackslash pdv[sep-eval-sp=\textbackslash here]\{f\}\{x,y\}\^{}\{(x\_2,y\_2)\}} \quad \Longrightarrow \quad \left(\frac{\partial^2 f}{\partial x \, \partial y} \right)^{\langle \texttt{here} \rangle (x_2, y_2)}$$

Switches

<code>switch-*</code>   `true, false`   `false`

The effect of the star argument can be toggled with the value `true`. For example, compare below where the option is disabled (`false`) and enabled (`true`). The default value is `false`.

$$\texttt{\textbackslash pdv[switch-*=false]\{f\}\{x,y\}} \quad \Longrightarrow \quad \frac{\partial^2 f}{\partial x \, \partial y}$$

$$\texttt{\textbackslash pdv[switch-*=true]\{f\}\{x,y\}} \quad \Longrightarrow \quad \frac{\partial^2}{\partial x \, \partial y} f$$

<code>switch-/</code>   `true, false`   `false`

The effect of the slash argument can be toggled with the value `true`. For example, compare below where the option is disabled (`false`) and enabled (`true`). The default value is `false`.

$$\texttt{\textbackslash pdv[switch-/=false]\{f\}\{x,y\}} \quad \Longrightarrow \quad \frac{\partial^2 f}{\partial x \, \partial y}$$

$$\texttt{\textbackslash pdv[switch-/=true]\{f\}\{x,y\}} \quad \Longrightarrow \quad \partial^2 f / \partial x \, \partial y$$

<code>switch-!</code>   `true, false`   `false`

New: v1.3   The effect of the exclamation mark argument can be toggled with the value `true`. For example, compare below where the option is disabled (`false`) and enabled (`true`). The default value is `false`.

$$\texttt{\textbackslash pdv[switch-!=false]\{f\}\{x,y\}} \quad \Longrightarrow \quad \frac{\partial^2 f}{\partial x \, \partial y}$$

$$\texttt{\textbackslash pdv[switch-!=true]\{f\}\{x,y\}} \quad \Longrightarrow \quad \partial_x \, \partial_y f$$

<code>switch-sort</code>   `true, false`   `true`

New: v1.2   This options disables (`false`) and enables (`true`) the sorting algorithm behind the mixed order. The sorting algorithm is only used/available in fraction mode when it is `multiple`. When disabled, the terms in the mixed order are arranged according to their first appearance in `order={…}`. Compare below where the option is disabled (`false`) and enabled (`true`). The default value is `true`.

$$\texttt{\textbackslash pdv[switch-sort=false, order=\{a+b,2kn-d,2-2b\}]\{f\}\{x,y,z\}} \quad \Longrightarrow \quad \frac{\partial^{a-b+2kn-d+2} f}{\partial x^{a+b} \, \partial y^{2kn-d} \, \partial z^{2-2b}}$$

$$\texttt{\textbackslash pdv[switch-sort=true, order=\{a+b,2kn-d,2-2b\}]\{f\}\{x,y,z\}} \quad \Longrightarrow \quad \frac{\partial^{2kn+a-b-d+2} f}{\partial x^{a+b} \, \partial y^{2kn-d} \, \partial z^{2-2b}}$$

21

Sort

The keys mentioned in this subsection will be briefly described here, with a more detailed explanation provided in section 6.

sort-method	abs, lexical, number, sign, symbol	sign, symbol, abs

Updated: v1.2

Sets the sorting method for the mixed order using build-in algorithms.

abs Sorts terms in descending order based on their absolute value.

lexical Sorts terms alphabetically in lexicographical ascending order.

number Sorts terms in descending order based on their numerical value.

sign Sorts terms by their sign, placing positive terms before the negative terms.

symbol Sorts terms in descending order based on their symbolic length.

This option accepts a comma-separated list of values, allowing up to three values. For example:

$$\text{\pdv[sort-method=abs, order=\{kn-c,2a-3b\}]\{f\}\{x,y\}} \implies \frac{\partial^{-3b+2a+kn-c} f}{\partial x^{kn-c} \partial y^{2a-3b}}$$

$$\text{\pdv[sort-method=lexical, order=\{kn-c,2a-3b\}]\{f\}\{x,y\}} \implies \frac{\partial^{2a-3b-c+kn} f}{\partial x^{kn-c} \partial y^{2a-3b}}$$

$$\text{\pdv[sort-method=number, order=\{kn-c,2a-3b\}]\{f\}\{x,y\}} \implies \frac{\partial^{2a+kn-c-3b} f}{\partial x^{kn-c} \partial y^{2a-3b}}$$

$$\text{\pdv[sort-method=symbol, order=\{kn-c,2a-3b\}]\{f\}\{x,y\}} \implies \frac{\partial^{kn-c+2a-3b} f}{\partial x^{kn-c} \partial y^{2a-3b}}$$

$$\text{\pdv[sort-method=sign, order=\{kn-c,2a-3b\}]\{f\}\{x,y\}} \implies \frac{\partial^{kn+2a-c-3b} f}{\partial x^{kn-c} \partial y^{2a-3b}}$$

$$\text{\pdv[sort-method=\{sign,abs\}, order=\{kn-c,2a-3b\}]\{f\}\{x,y\}} \implies \frac{\partial^{2a+kn-3b-c} f}{\partial x^{kn-c} \partial y^{2a-3b}}$$

$$\text{\pdv[sort-method=\{sign,symbol,abs\}, order=\{kn-c,2a-3b\}]\{f\}\{x,y\}} \implies \frac{\partial^{kn+2a-3b-c} f}{\partial x^{kn-c} \partial y^{2a-3b}}$$

Notice the different term ordering achieved with each method. For a more detailed explanation of this key, see section 6.1 for more information. The default value uses the three algorithms `sign`, `symbol`, `abs`.

sort-numerical	auto, first, last, symbolic	auto

Updated: v1.0

Determines the placement of the numerical term[2] in the mixed order. The placement options are as follows:

---

[2]The numerical term refers to the sum of all orders that consist solely of numbers and do not include any symbols.

first Always places the numerical term as the first term in the mixed order.

last Always places the numerical term as the last term in the mixed order.

auto After sorting, the numerical term is automatically positioned based on the sign of the first symbolic term. If positive, the numerical term is placed last. If negative, the numerical term is placed first.

symbolic Treats the numerical term as a symbolic term with a symbolic length of zero and sorts it alongside all other terms.

Here are some examples to illustrate the different placements of the numerical term:

$$\text{\\pdv[sort-numerical=first, order=\{n,2\}]\{f\}\{x,y\}} \implies \frac{\partial^{2+n} f}{\partial x^n \, \partial y^2}$$

$$\text{\\pdv[sort-numerical=last, order=\{-n,2\}]\{f\}\{x,y\}} \implies \frac{\partial^{-n+2} f}{\partial x^{-n} \, \partial y^2}$$

$$\text{\\pdv[sort-numerical=auto, order=\{n,2\}]\{f\}\{x,y\}} \implies \frac{\partial^{n+2} f}{\partial x^n \, \partial y^2}$$

$$\text{\\pdv[sort-numerical=auto, order=\{-n,2\}]\{f\}\{x,y\}} \implies \frac{\partial^{2-n} f}{\partial x^{-n} \, \partial y^2}$$

$$\text{\\pdv[sort-numerical=symbolic, order=\{2+n,-a\}]\{f\}\{x,y\}} \implies \frac{\partial^{n+2-a} f}{\partial x^{2+n} \, \partial y^{-a}}$$

This key is further described in section 6.2. The default value is `auto`.

`sort-abs-reverse`   true, false   false

When the value `true` is used with this option, the sorting order is reversed to ascending order. See section 6.3 for more information.

$$\text{\\pdv[sort-abs-reverse=false, sort-method=abs, order=2a-3b]\{f\}\{x\}} \implies \frac{\partial^{-3b+2a} f}{\partial x^{2a-3b}}$$

$$\text{\\pdv[sort-abs-reverse=true, sort-method=abs, order=2a-3b]\{f\}\{x\}} \implies \frac{\partial^{2a-3b} f}{\partial x^{2a-3b}}$$

`sort-lexical-reverse`   true, false   false

When the value `true` is used with this option, the sorting order is reversed to lexicographical descending order. See section 6.3 for more information.

$$\text{\\pdv[sort-lexical-reverse=false, sort-method=lexical, order=a+b]\{f\}\{x\}} \implies \frac{\partial^{a+b} f}{\partial x^{a+b}}$$

$$\text{\\pdv[sort-lexical-reverse=true, sort-method=lexical, order=a+b]\{f\}\{x\}} \implies \frac{\partial^{b+a} f}{\partial x^{a+b}}$$

When the value `true` is used with this option, the sorting order is reversed to ascending order. See section 6.3 for more information.

$$\texttt{\textbackslash pdv[sort-number-reverse=false, sort-method=number, order=2a-3b]\{f\}\{x\}} \implies \frac{\partial^{2a-3b} f}{\partial x^{2a-3b}}$$

$$\texttt{\textbackslash pdv[sort-number-reverse=true, sort-method=number, order=2a-3b]\{f\}\{x\}} \implies \frac{\partial^{-3b+2a} f}{\partial x^{2a-3b}}$$

sort-sign-reverse | true, false | false

When the value `true` is used with this option, the sorting order is reversed, placing negative terms before the positive terms. See section 6.3 for more information.

$$\texttt{\textbackslash pdv[sort-sign-reverse=false, sort-method=sign, order=a-b]\{f\}\{x\}} \implies \frac{\partial^{a-b} f}{\partial x^{a-b}}$$

$$\texttt{\textbackslash pdv[sort-sign-reverse=true, sort-method=sign, order=a-b]\{f\}\{x\}} \implies \frac{\partial^{-b+a} f}{\partial x^{a-b}}$$

sort-symbol-reverse | true, false | false

When the value `true` is used with this option, the sorting order is reversed to ascending order. See section 6.3 for more information.

$$\texttt{\textbackslash pdv[sort-symbol-reverse=false, sort-method=symbol, order=ab+c]\{f\}\{x\}} \implies \frac{\partial^{ab+c} f}{\partial x^{ab+c}}$$

$$\texttt{\textbackslash pdv[sort-symbol-reverse=true, sort-method=symbol, order=ab+c]\{f\}\{x\}} \implies \frac{\partial^{c+ab} f}{\partial x^{ab+c}}$$

Miscellaneous

fun | true, false | false

New: v1.0

This option allows you to add (`true`) or remove (`false`) delimiters around the function. Note that if the option is not explicitly set, it is considered equivalent to `fun=true`. By default, the value is `false`.

$$\texttt{\textbackslash pdv[fun=false]\{f\}\{x\}} \implies \frac{\partial f}{\partial x}$$

$$\texttt{\textbackslash pdv[fun=true]\{f\}\{x\}} \implies \frac{\partial (f)}{\partial x}$$

$$\texttt{\textbackslash pdv[fun]\{f\}\{x\}} \implies \frac{\partial (f)}{\partial x}$$

frac | true, false | false

New: v1.0

This option allows you to add (`true`) or remove (`false`) delimiters around the fractional part of the derivative. Note that if the option is not explicitly set, it is considered equivalent to `frac=true`. By default, the value is `false`.

$$\text{\pdv[frac=false]\{f\}\{x\}} \implies \frac{\partial f}{\partial x}$$

$$\text{\pdv[frac=true]\{f\}\{x\}} \implies \left(\frac{\partial f}{\partial x}\right)$$

$$\text{\pdv*[frac=true]\{f\}\{x\}} \implies \left(\frac{\partial}{\partial x}\right)f$$

$$\text{\pdv[frac]\{f\}\{x\}} \implies \left(\frac{\partial f}{\partial x}\right)$$

var	none, all, ⟨cs-numbers⟩	none

New: v1.0

This option allows you to add or remove delimiters around the variables. The value `all` adds delimiters around all variables and `none` removed all delimiters. If only specific variables require delimiters, you can use ⟨cs-numbers⟩. For example, `var={1,3}` adds delimiters around the first and third variables. Note that if the option is not explicitly set, it is considered equivalent to `var=all`. By default, the value is `none`.

$$\text{\pdv[var=none]\{f\}\{x,y,z,t\}} \implies \frac{\partial^4 f}{\partial x\,\partial y\,\partial z\,\partial t}$$

$$\text{\pdv[var=\{1,3\}]\{f\}\{x,y,z,t\}} \implies \frac{\partial^4 f}{\partial(x)\,\partial y\,\partial(z)\,\partial t}$$

$$\text{\pdv[var=all]\{f\}\{x,y,z,t\}} \implies \frac{\partial^4 f}{\partial(x)\,\partial(y)\,\partial(z)\,\partial(t)}$$

$$\text{\pdv[var]\{f\}\{x,y,z,t\}} \implies \frac{\partial^4 f}{\partial(x)\,\partial(y)\,\partial(z)\,\partial(t)}$$

order	{⟨cs-orders⟩}	1
ord		

New: v1.0

Sets the order of differentiation for each variable as a comma separated list of values.

$$\text{\pdv[order=\{2,3\}]\{f\}\{x,y,z\}} \implies \frac{\partial^6 f}{\partial x^2\,\partial y^3\,\partial z}$$

$$\text{\pdv[order=\{\beta,a,n+2a\}]\{f\}\{x,y,z\}} \implies \frac{\partial^{3a+\beta+n} f}{\partial x^\beta\,\partial y^a\,\partial z^{n+2a}}$$

$$\text{\pdv[order=\{2,n\^{}2,n\^{}2-1\}]\{f\}\{x,y,z\}} \implies \frac{\partial^{2n^2+1} f}{\partial x^2\,\partial y^{n^2}\,\partial z^{n^2-1}}$$

$$\text{\pdv[order=\{3/2-n/3,n/2,1/3\}]\{f\}\{x,y,z\}} \implies \frac{\partial^{n/6+11/6} f}{\partial x^{3/2-n/3}\,\partial y^{n/2}\,\partial z^{1/3}}$$

mixed-order	{⟨mixed order⟩}	1
mixord		

New: v1.0

This option allows you to manually specify the mixed order of differentiation, overriding the automatically calculated value based on the orders set by `order={⟨orders⟩}`. In cases where the automatic calculation fails or when a different form is preferred, you can use this option to set the desired mixed order.

$$\texttt{\textbackslash pdv[order=\{n+3k, n-k\}]\{f\}\{x,y\}} \quad \Longrightarrow \quad \frac{\partial^{2n+2k} f}{\partial x^{n+3k} \, \partial y^{n-k}}$$

$$\texttt{\textbackslash pdv[order=\{n+3k, n-k\}, mixed-order=\{2(n+k)\}]\{f\}\{x,y\}} \quad \Longrightarrow \quad \frac{\partial^{2(n+k)} f}{\partial x^{n+3k} \, \partial y^{n-k}}$$

4.4 Differential options

The options in this subsection are available for customizing the behaviour of differentials defined by the package and you.

Style

style-inf $\langle\textit{math-font-style}\rangle\langle\textit{infinitesimal}\rangle$ d

Sets the infinitesimal used in the differential. The default infinitesimal is a plain d.

$$\texttt{\textbackslash odif[style-inf=\textbackslash mathbf\{d\}]\{x\_1,x\_2,x\_3\}} \quad \Longrightarrow \quad \mathbf{d}x_1 \, \mathbf{d}x_2 \, \mathbf{d}x_3$$

style-var   single, multiple, mixed   multiple

Updated: v1.3   This options determines the shorthand notation of the differential when the star argument is absent assuming `switch-*=false`.

single   Creates a single differential with variables as subscripts and orders as superscripts.

multiple   Creates a separate differential for each variable.

mixed   Creates a separate differential for each variable but with variables as subscripts and orders as superscripts.

$$\texttt{\textbackslash odif[style-var=multiple, order=\{n,1,2\}]\{x,y,z,t\}} \quad \Longrightarrow \quad \mathrm{d}^n x \, \mathrm{d}y \, \mathrm{d}^2 z \, \mathrm{d}t$$

$$\texttt{\textbackslash odif[style-var=single, order=\{n,1,2\}]\{x,y,z,t\}} \quad \Longrightarrow \quad \mathrm{d}^{n,1,2}_{x,y,z,t}$$

$$\texttt{\textbackslash odif[style-var=mixed, order=\{n,1,2\}]\{x,y,z,t\}} \quad \Longrightarrow \quad \mathrm{d}^n_x \, \mathrm{d}_y \, \mathrm{d}^2_z \, \mathrm{d}_t$$

style-var-*   single, multiple, mixed   single

Updated: v1.3   This option behaves identically to `style-var`, however it is used when the star argument is present assuming `switch-*=false`.

Scaling

scale-var   auto, none, big, Big, bigg, Bigg   auto

Sets the size of the delimiters used around the variables specified by the `var={…}` option when the star argument is absent assuming `switch-*=false`. The default behaviour is to automatically scale the delimiters.

$$\text{\odif[scale-var=none, var]\{r,\theta,\varphi\}} \quad \implies \quad \mathrm{d}(r)\,\mathrm{d}(\theta)\,\mathrm{d}(\varphi)$$

$$\text{\odif[scale-var=Big, var]\{r,\theta,\varphi\}} \quad \implies \quad \mathrm{d}\Big(r\Big)\mathrm{d}\Big(\theta\Big)\mathrm{d}\Big(\varphi\Big)$$

`scale-var-*`   `auto, none, big, Big, bigg, Bigg`   `auto`

Sets the size of the delimiters used around the variables specified by the `var={…}` option when the star argument is present assuming `switch-*=false`. The default behaviour is to automatically scale the delimiters.

$$\text{\odif[scale-var-*=auto, var]\{r,\theta,\varphi\}} \quad \implies \quad \mathrm{d}_{(r),(\theta),(\varphi)}$$

$$\text{\odif[scale-var-*=big, var]\{r,\theta,\varphi\}} \quad \implies \quad \mathrm{d}_{\big(r\big),\big(\theta\big),\big(\varphi\big)}$$

Delimiters

`delims-var`   *⟨left delimiter⟩⟨right delimiter⟩*   `( )`

Sets the delimiters used around the variables specified by the `var={…}` option when the star argument is absent assuming `switch-*=false`. The default delimiters are parentheses.

$$\text{\odif[delims-var=\\\{\\\}, var]\{x,y\}} \quad \implies \quad \mathrm{d}\{x\}\,\mathrm{d}\{y\}$$

$$\text{\odif[delims-var=[], var]\{x,y\}} \quad \implies \quad \mathrm{d}[x]\,\mathrm{d}[y]$$

`delims-var-*`   *⟨left delimiter⟩⟨right delimiter⟩*   `( )`

Sets the delimiters used around the variables specified by the `var={…}` option when the star argument is present assuming `switch-*=false`. The default delimiters are parentheses.

$$\text{\odif[delims-var-*=\\|\\|, var]\{x,y\}} \quad \implies \quad \mathrm{d}_{\|x\|,\|y\|}$$

$$\text{\odif[delims-var-*=\\langle\\rangle, var]\{x,y\}} \quad \implies \quad \mathrm{d}_{\langle x\rangle,\langle y\rangle}$$

Math spacing and separators

`sep-begin`   *⟨cs-number⟩*, {*⟨mspace⟩*}, {*⟨delimiter⟩*}   `\mathop{}\!`

Sets the separator that is inserted at the beginning of the expression. The default value is `\mathop{}\!`.

$$\text{\odif[sep-begin=\\here]\{x,y,z\}} \quad \implies \quad _{\langle\text{here}\rangle}\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z$$

`sep-inf-ord`   *⟨cs-number⟩*, {*⟨mspace⟩*}, {*⟨delimiter⟩*}   `0`

Sets the separator that is inserted to the left of the infinitesimal's power when the order *is different* from 1. The default value corresponds to `0 mu`.

$$\text{\odif[sep-inf-ord=\\here, order=2]\{x,y\}} \quad \implies \quad \mathrm{d}^{\langle\text{here}\rangle 2}x\,\mathrm{d}y$$

`sep-inf-var`   ⟨*cs-number*⟩, {⟨*mspace*⟩}, {⟨*delimiter*⟩}   0

Sets the separator that is inserted between the infinitesimal and the following non-blank variable when the associated order *is equal* to 1. The key's default value corresponds to `0 mu`.

$$\texttt{\textbackslash odif[sep-inf-var=\textbackslash here, order=2]\{x,y\}} \quad \Longrightarrow \quad \mathrm{d}^2 x\,\mathrm{d}_{\langle\texttt{here}\rangle}y$$

`sep-ord-var`   ⟨*cs-number*⟩, {⟨*mspace*⟩}, {⟨*delimiter*⟩}   0

Sets the separator that is inserted between an order and the following non-blank variable when the order *is different* from 1. This option is only available when the style is set to `multiple`. The default value corresponds to `0 mu`.

$$\texttt{\textbackslash odif[sep-ord-var=\textbackslash here, order=2]\{x,y\}} \quad \Longrightarrow \quad \mathrm{d}^2{}_{\langle\texttt{here}\rangle}x\,\mathrm{d}y$$

`sep-var-inf`   ⟨*cs-number*⟩, {⟨*mspace*⟩}, {⟨*delimiter*⟩}   `\mathop{}\!`

Sets the separator that is inserted between a non-blank variable and the following infinitesimal when more than one non-blank variables are given. The default value is `\mathop{}\!`.

$$\texttt{\textbackslash odif[sep-var-inf=\textbackslash here]\{x,y\}} \quad \Longrightarrow \quad \mathrm{d}x_{\langle\texttt{here}\rangle}\mathrm{d}y$$

`sep-var-var`   ⟨*cs-number*⟩, {⟨*mspace*⟩}, {⟨*delimiter*⟩}   ,

Sets the separator that is inserted between two variables when more than one non-blank variables are given. This option is only available when the style is set to `single`. The default value is a comma.

$$\texttt{\textbackslash odif*[sep-var-var=\textbackslash here]\{x,y\}} \quad \Longrightarrow \quad \mathrm{d}_{x_{\langle\texttt{here}\rangle}y}$$

`sep-ord-ord`   ⟨*cs-number*⟩, {⟨*mspace*⟩}, {⟨*delimiter*⟩}   ,

Sets the separator that is inserted between two orders that are both not equal to 1. This option is only available when the style is set to `single`. The default value is a comma.

$$\texttt{\textbackslash odif*[sep-ord-ord=\textbackslash here, order=\{2,n\}]\{x,y\}} \quad \Longrightarrow \quad \mathrm{d}_{x,y}^{2_{\langle\texttt{here}\rangle}n}$$

`sep-end`   ⟨*cs-number*⟩, {⟨*mspace*⟩}, {⟨*delimiter*⟩}   0

Sets the separator that is inserted at the end of the expression. The default value corresponds to `0 mu`.

$$\texttt{\textbackslash odif[sep-end=\textbackslash here]\{x,y\}} \quad \Longrightarrow \quad \mathrm{d}x\,\mathrm{d}y_{\langle\texttt{here}\rangle}$$

Switches

switch-*   true, false   false

The effect of the star argument can be toggled with the value `true`. For example, compare below where the option is disabled (`false`) and enabled (`true`). The default value is `false`.

$$\odif[\texttt{switch-*=false}]\{x,y,z,t\} \implies \mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z\,\mathrm{d}t$$

$$\odif[\texttt{switch-*=true}]\{x,y,z,t\} \implies \mathrm{d}_{x,y,z,t}$$

Miscellaneous

var   none, all, ⟨*cs-numbers*⟩   none

This option allows you to add or remove delimiters around the variables. The value `all` adds delimiters around all variables and `none` removed all delimiters. If only specific variables require delimiters, you can use ⟨*cs-numbers*⟩. For example, `var={1,3}` adds delimiters around the first and third variables. Note that if the option is not explicitly set, it is considered equivalent to `all`. By default, the value is `none`.

$$\odif[\texttt{var=none}]\{x,y,z,t\} \implies \mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z\,\mathrm{d}t$$

$$\odif[\texttt{var=\{1,3\}}]\{x,y,z,t\} \implies \mathrm{d}(x)\,\mathrm{d}y\,\mathrm{d}(z)\,\mathrm{d}t$$

$$\odif[\texttt{var=all}]\{x,y,z,t\} \implies \mathrm{d}(x)\,\mathrm{d}(y)\,\mathrm{d}(z)\,\mathrm{d}(t)$$

$$\odif[\texttt{var}]\{x,y,z,t\} \implies \mathrm{d}(x)\,\mathrm{d}(y)\,\mathrm{d}(z)\,\mathrm{d}(t)$$

order   {⟨*cs-orders*⟩}   1

ord   Sets the order of differentiation for each variable as a comma separated list of values.

$$\odif[\texttt{order=\{2,3\}}]\{x,y,z\} \implies \mathrm{d}^2x\,\mathrm{d}^3y\,\mathrm{d}z$$

$$\odif[\texttt{order=\{\textbackslash beta,a,n+2a\}}]\{x,y,z\} \implies \mathrm{d}^\beta x\,\mathrm{d}^a y\,\mathrm{d}^{n+2a}z$$

4.5 All derivatives and differentials

The options in this subsection are applied to all derivatives and differentials because they define settings that should be consistent regardless of the specific derivative or differential. These options can be accessed using the command \derivset{all}[⟨*key=value*⟩].

The options in this subsection are applied to all derivatives and differential because some options should be consisting regardless of the derivative and differential. The options are accessed using \derivset{all}[⟨*key=value*⟩].

$$\derivset\{all\}[key=value]$$

scale-auto   leftright, mleftmright[R]   leftright or mleftmright[R]

    Sets the automatic delimiter scaling commands. The value `leftright` sets them as \left and \right while `mleftmright` sets them as \mleft and \mright from the `mleftrigth` package. The default value is `leftright` unless `mleftright` is loaded, in which case it is `mleftmright`.

`\derivset{all}[scale-auto=leftright] a \pdv[frac]{f}{x} a` $\implies a \left(\dfrac{\partial f}{\partial x} \right) a$

`\derivset{all}[scale-auto=mleftmright] a \pdv[frac]{f}{x} a` $\implies a\left(\dfrac{\partial f}{\partial x}\right)a$

Notice the space difference between the a's and the parentheses in the two equations.

5 Defining variants

This section goes into detail on how to define derivative and differential variants based on the package's internal commands. The derivative package provides a LaTeX 2$_\varepsilon$-based approach to defining derivatives and differentials.

5.1 Derivative variant

\NewDerivative
\RenewDerivative
\ProvideDerivative
\DeclareDerivative

⟨*control-sequence*⟩{⟨*infinitesimal*⟩}[⟨*key=value*⟩]
This family of commands is used to define a derivative variant with the macro name ⟨*control-sequence*⟩.

- \NewDerivative will issue an error if ⟨*control-sequence*⟩ has already been defined.

- \RenewDerivative will issue an error if ⟨*control-sequence*⟩ has not previously been defined.

- \ProvideDerivative will define ⟨*control-sequence*⟩ if it does not have an existing definition. It will not issue any errors.

- \DeclareDerivative will always define the ⟨*control-sequence*⟩ with the new definition regardless of whether it already exists.

Examples of their use can be found in section 2.

⟨*control-sequence*⟩   The first argument specifies the macro name of the derivative being defined.

{⟨*infinitesimal*⟩}   The second argument sets the infinitesimal of the derivative ⟨*control-sequence*⟩. It is equivalent to setting style-inf={⟨*infinitesimal*⟩}.

[⟨*key=value*⟩]   The optional argument accepts a comma-separated list of *key=value* pairs, allowing you to override the package's default values for the keys given. If the optional argument is omitted, the derivative will use the package's default settings.

5.2 Differential variant

\NewDifferential
\RenewDifferential
\ProvideDifferential
\DeclareDifferential

⟨*control-sequence*⟩{⟨*infinitesimal*⟩}[⟨*key=value*⟩]
This family of commands is used to define a differential variant with the macro name ⟨*control-sequence*⟩.

- \NewDifferential will issue an error if ⟨*control-sequence*⟩ has already been defined.

- \RenewDifferential will issue an error if ⟨*control-sequence*⟩ has not previously been defined.

- \ProvideDifferential will define ⟨*control-sequence*⟩ if it does not have an existing definition. It will not issue any errors.

- \DeclareDifferential will always define the ⟨*control-sequence*⟩ with the new definition regardless of whether it already exists.

Examples of their use can be found in section 3.

⟨*control-sequence*⟩ The first argument specifies the macro name of the differential being defined.

{⟨*infinitesimal*⟩} The second argument sets the infinitesimal of the differential ⟨*control-sequence*⟩. It is equivalent to setting `style-inf={`⟨*infinitesimal*⟩`}`.

[⟨*key=value*⟩] The optional argument accepts a comma-separated list of `{key=value}` pairs, allowing you to override the package's default values for the keys given. If the optional argument is omitted, the derivative will use the package's default settings.

6 The mixed order

The algorithm used to calculate the mixed order is advanced and can handle a wide range of inputs, including numbers, symbols and fractions. It accurately parses the orders to generate the appropriate mixed order. Let's consider some examples to illustrate its capabilities:

$$\text{\pdv[order={n+1,2}]{f}{x,y}} \implies \frac{\partial^{n+3} f}{\partial x^{n+1} \, \partial y^2}$$

$$\text{\pdv[order={n+1,2n}]{f}{x,y}} \implies \frac{\partial^{3n+1} f}{\partial x^{n+1} \, \partial y^{2n}}$$

$$\text{\pdv[order={n\^{}2,n\^{}2}]{f}{x,y}} \implies \frac{\partial^{2n^2} f}{\partial x^{n^2} \, \partial y^{n^2}}$$

$$\text{\pdv[order={2/n-1/2, 3/n+m/3}]{f}{x,y,z}} \implies \frac{\partial^{m/3+5/n+1/2} f}{\partial x^{2/n-1/2} \, \partial y^{3/n+m/3} \, \partial z}$$

The algorithm handles these cases and calculates the mixed order correctly. However, there are certain limitations when dealing with parentheses, except for a specific case where there is a single term in the denominator.

$$\text{\pdv[order={2/n-1/2, 3/(2n)+m/3}]{f}{x,y}} \implies \frac{\partial^{7/(2n)+m/3-1/2} f}{\partial x^{2/n-1/2} \, \partial y^{3/(2n)+m/3}}$$

$$\text{\pdv[order={(2n+m)/3, 3n+m/3}]{f}{x,y}} \implies \frac{\partial^{m)/3+(2n+m/3+3n} f}{\partial x^{(2n+m)/3} \, \partial y^{3n+m/3}}$$

$$\text{\pdv[order={2/(2n+m), 1/(2(2n+m))}]{f}{x,y}} \implies \frac{\partial^{1/(2(2n+2/(2n+m))+m)} f}{\partial x^{2/(2n+m)} \, \partial y^{1/(2(2n+m))}}$$

In the third equation, the mixed order doesn't make sense because the algorithm splits up the orders at $+$ and $-$ within parentheses without evaluating them correctly. For example, $2/(2n+m), 1/(2(2n+m))$ is incorrectly split into $2/(2n, m), 1/(2(2n, m))$. The same happens in the second equation. To address this limitation, the package provides the `mixed-order` option, which overrides the incorrect output. You can specify the desired mixed order manually.

$$\text{\pdv[order={(2n+m)/3, 3n+m/3}, mixord=8n/3+2m/3]{f}{x,y}} \implies \frac{\partial^{8n/3+2m/3} f}{\partial x^{(2n+m)/3} \, \partial y^{3n+m/3}}$$

$$\text{\pdv[order={2/(2n+m), 1/(2(2n+m))}, mixord=5/(2(2n+m))]{f}{x,y}} \implies \frac{\partial^{5/(2(2n+m))} f}{\partial x^{2/(2n+m)} \, \partial y^{1/(2(2n+m))}}$$

By using the `mixed-order` option, you can ensure that the mixed order is displayed correctly, even when the algorithm's output may be incorrect or when a different form is preferred.

6.1 Sorting algorithms

A unique feature of this package is that the sorting method behind the mixed order may be changed using built-in algorithms. These algorithms are designed to offer variety of ways to sort the mixed order according to ones preferences.

abs Sorts terms based on their absolute value in descending order. By setting `sort-abs-reverse=true`, terms are sorted in ascending order instead.

lexical Sorts the terms alphabetically in lexicographical order. By setting `sort-lexical-reverse=true`, terms are sorted in reverse lexicographical order.

number Sorts terms based on their numerical value in descending order. By setting `sort-number-reverse=true`, terms are sorted in ascending order instead.

sign Sorts terms by their sign, placing positive terms before negative terms. By setting, `sort-sign-reverse=true`, places negative terms before positive terms.

symbol Sorts terms in descending order based on their symbolic length. By setting `sort-symbol-reverse=true`, terms are sorted in ascending order instead.

The sorting method allows up to three algorithms given by `sort-method={algorithm1, algorithm2, algorithm3}`. The list of orders is primarily sorted based on algorithm 1's condition. In case of ties, algorithm 2 is used for further sorting, and so on.

Consider the list `{a,-c,-b,d}` and the sorting method `sort-method={sign,lexical}`. Initially, the `sign` algorithm sorts the list as `{a,d,-c,-b}`. However, due to ties, other valid orderings exist like `{d,a,-c,-b}`, `{a,d,-b,-c}` and `{d,a,-b,-c}`. Subsequently, the `lexical` algorithm sorts the tied values alphabetically as `{a,d,-b,-c}`.

6.1.1 Examples

The examples below use partial derivatives with `order={3a-3hh-2b, 4c+4gg+2ff, -5d-5ee}`, along with a close-up view of the mixed order. Square brackets indicate how the terms was sorted by the applied algorithm. 'Positive' and 'negative', 'long' and 'short', and 'big' and 'low' refer to the `sign`, `symbol`, and `abs` algorithms, respectively.

The default sorting method is `sort-method={sign,symbol,abs}` which is used below. The `sign` algorithm separate the positive and negative terms. Within each group, the `symbol` algorithm sorts the terms based on symbolic length. Finally, In the resulting four groups, the `abs` algorithm sorts the terms based on absolute value.

$$\frac{\partial^{4gg+2ff+4c+3a-5ee-3hh-5d-2b} f}{\partial x^{3a-3hh-2b} \, \partial y^{4c+4gg+2ff} \, \partial z^{-5d-5ee}}$$

$$\underbrace{\underbrace{\underbrace{4gg}_{\text{big}} + \underbrace{2ff}_{\text{low}}}_{\text{long}} + \underbrace{\underbrace{4c}_{\text{big}} + \underbrace{3a}_{\text{low}}}_{\text{short}}}_{\text{positive}} - \underbrace{\underbrace{\underbrace{5ee}_{\text{big}} - \underbrace{3hh}_{\text{low}}}_{\text{long}} - \underbrace{\underbrace{5d}_{\text{big}} - \underbrace{2b}_{\text{low}}}_{\text{short}}}_{\text{negative}}$$

34

By interchanging the `sign` and `symbol` algorithm in the previous example, then the `symbol` algorithm will be applied first and the `sign` algorithm handles the symbolic ties.

$$\frac{\partial^{4gg+2ff-5ee-3hh+4c+3a-5d-2b} f}{\partial x^{3a-3hh-2b}\, \partial y^{4c+4gg+2ff}\, \partial z^{-5d-5ee}}$$

$$\underbrace{\underbrace{4gg}_{\text{big}} + \underbrace{2ff}_{\text{low}} - \underbrace{\underbrace{5ee}_{\text{big}} - \underbrace{3hh}_{\text{low}}}_{\text{negative}} + \underbrace{\underbrace{4c}_{\text{big}} + \underbrace{3a}_{\text{low}}}_{\text{positive}} - \underbrace{\underbrace{5d}_{\text{big}} - \underbrace{2b}_{\text{low}}}_{\text{negative}}}_{}$$
$$\underbrace{}_{\text{long}}\ \underbrace{}_{\text{short}}$$

In contrast to the previous two examples with three algorithms, this example uses a sorting method with two algorithms: `sort-method={sign,symbol}`. The terms are first sorted by sign and then by symbolic length.

$$\frac{\partial^{4gg+2ff+3a+4c-3hh-5ee-2b-5d} f}{\partial x^{3a-3hh-2b}\, \partial y^{4c+4gg+2ff}\, \partial z^{-5d-5ee}}$$

$$\underbrace{\underbrace{4gg+2ff}_{\text{long}} + \underbrace{3a+4c}_{\text{short}}}_{\text{positive}} - \underbrace{\underbrace{3hh-5ee}_{\text{long}} - \underbrace{2b-5d}_{\text{short}}}_{\text{negative}}$$

In the last example, a sorting method with a single algorithm, `sort-method=symbol`, is used.

$$\frac{\partial^{-3hh+4gg+2ff-5ee+3a-2b+4c-5d} f}{\partial x^{3a-3hh-2b}\, \partial y^{4c+4gg+2ff}\, \partial z^{-5d-5ee}}$$

$$\underbrace{-3hh+4gg+2ff-5ee}_{\text{long}} + \underbrace{3a-2b+4c-5d}_{\text{short}}$$

6.2 Treating the numerical term

After handling the symbolic part of the mixed order, we now turn our attention to the numerical term[2], which has a symbolic length of zero. For this reason it is treated differently than the symbolic terms. It can be placed either at the beginning or at the end of the mixed order by `first` and `last`, respectively. Alternatively, it is automatically positioned in the mixed order using `auto`. Setting `sort-numerical=symbolic`, treats the numerical term as a symbolic term. Its behaviour is determined by the chosen algorithms in `sort-method={…}`. In this example the sort method is the default value of the package.

$$\texttt{sort-numerical=auto:} \qquad \frac{\partial^{\beta+2} f}{\partial x^{\beta}\, \partial y^{2}} \qquad \frac{\partial^{\beta-2} f}{\partial x^{\beta}\, \partial y^{-2}} \qquad \frac{\partial^{2-\beta} f}{\partial x^{-\beta}\, \partial y^{2}} \qquad \frac{\partial^{-2-\beta} f}{\partial x^{-\beta}\, \partial y^{-2}}$$

$$\texttt{sort-numerical=first:} \qquad \frac{\partial^{2+\beta} f}{\partial x^{\beta}\, \partial y^{2}} \qquad \frac{\partial^{-2+\beta} f}{\partial x^{\beta}\, \partial y^{-2}} \qquad \frac{\partial^{2-\beta} f}{\partial x^{-\beta}\, \partial y^{2}} \qquad \frac{\partial^{-2-\beta} f}{\partial x^{-\beta}\, \partial y^{-2}}$$

$$\texttt{sort-numerical=last:} \qquad \frac{\partial^{\beta+2} f}{\partial x^{\beta}\, \partial y^{2}} \qquad \frac{\partial^{\beta-2} f}{\partial x^{\beta}\, \partial y^{-2}} \qquad \frac{\partial^{-\beta+2} f}{\partial x^{-\beta}\, \partial y^{2}} \qquad \frac{\partial^{-\beta-2} f}{\partial x^{-\beta}\, \partial y^{-2}}$$

$$\texttt{sort-numerical=symbolic:} \qquad \frac{\partial^{\beta+2} f}{\partial x^{\beta}\, \partial y^{2}} \qquad \frac{\partial^{\beta-2} f}{\partial x^{\beta}\, \partial y^{-2}} \qquad \frac{\partial^{2-\beta} f}{\partial x^{-\beta}\, \partial y^{2}} \qquad \frac{\partial^{-\beta-2} f}{\partial x^{-\beta}\, \partial y^{-2}}$$

6.3 Reversing the sort algorithm

The reverse keys serves to reverse the sorting algorithms' ordering, providing more flexibility over the sorting method. If the default ordering of an algorithm is not preferred it may be reversed with the corresponding reverse key. The examples below showcase the reverse functionality. Note that `sort-method={1 algorithm only}` is also used to better demonstrate the reverse feature, where the algorithm is the one used in the reverse key.

sort-abs-reverse=false: $\dfrac{\partial^{-3b+2a} f}{\partial x^{2a}\, \partial y^{-3b}}$ sort-abs-reverse=true: $\dfrac{\partial^{2a-3b} f}{\partial x^{2a}\, \partial y^{-3b}}$

sort-number-reverse=false: $\dfrac{\partial^{2a-3b} f}{\partial x^{2a}\, \partial y^{-3b}}$ sort-number-reverse=true: $\dfrac{\partial^{-3b+2a} f}{\partial x^{2a}\, \partial y^{-3b}}$

sort-sign-reverse=false: $\dfrac{\partial^{a-b} f}{\partial x^{a}\, \partial y^{-b}}$ sort-sign-reverse=true: $\dfrac{\partial^{-b+a} f}{\partial x^{a}\, \partial y^{-b}}$

sort-symbol-reverse=false: $\dfrac{\partial^{ab+c} f}{\partial x^{ab}\, \partial y^{c}}$ sort-symbol-reverse=true: $\dfrac{\partial^{c+ab} f}{\partial x^{ab}\, \partial y^{c}}$

sort-lexical-reverse=false: $\dfrac{\partial^{a+b+c} f}{\partial x^{a}\, \partial y^{c}\, \partial z^{b}}$ sort-lexical-reverse=true: $\dfrac{\partial^{c+b+a} f}{\partial x^{a}\, \partial y^{c}\, \partial z^{b}}$

7 Miscellaneous

7.1 Slashfrac

\slashfrac [⟨*scale*⟩]{⟨*numerator*⟩}{⟨*denominator*⟩}

The text-styled fraction, a/b, is commonly used in text-mode. While \slashfrac{a}{b} requires more keystrokes to write than a/b, a macro is needed for implementing text-styled derivatives.

[⟨*scale*⟩] The optional argument serves as the scaling parameter for the solidus and accepts the following inputs:

$$\texttt{\textbackslash slashfrac\{y\_f\}\{x\}} \quad \Longrightarrow \quad y_f/x$$

$$\texttt{\textbackslash slashfrac[auto]\{y\_f\}\{x\}} \quad \Longrightarrow \quad y_f/x$$

$$\texttt{\textbackslash slashfrac[none]\{y\_f\}\{x\}} \quad \Longrightarrow \quad y_f/x$$

$$\texttt{\textbackslash slashfrac[big]\{y\_f\}\{x\}} \quad \Longrightarrow \quad y_f\big/x$$

$$\texttt{\textbackslash slashfrac[Big]\{y\_f\}\{x\}} \quad \Longrightarrow \quad y_f\Big/x$$

$$\texttt{\textbackslash slashfrac[bigg]\{y\_f\}\{x\}} \quad \Longrightarrow \quad y_f\bigg/x$$

$$\texttt{\textbackslash slashfrac[Bigg]\{y\_f\}\{x\}} \quad \Longrightarrow \quad y_f\Bigg/x$$

Omitting [⟨*scale*⟩] sets the scaling parameter to auto.

{⟨*numerator*⟩} Typesets the fraction's numerator.

{⟨*denominator*⟩} Typesets the fraction's denominator.

Index

Numbers in bold refer to the page where the entry is defined.

Index of Options

Index of Commands

Change history

v0.9
2019-07-21
Beta

- First release of the package. The package is currently in a beta version.

v0.95
2019-09-18
Beta

- *Please ignore this version, since it contained the wrong* .sty *and* .pdf *files* :(.
- Removed the single token restriction on the infinitesimal since it was unnecessary.
- Fixed documentation errors and typos.
- Made minor fixes to the code.

v0.95b
2019-09-21
Beta

- Contains the correct .sty and .pdf files :).
- Made one minor code fix.

v0.96
2019-12-22
Beta

- Fixed the issue with double superscript for higher-order derivatives when the variable contained a superscript.

v0.97
2020-02-03
Beta

- Fixed the argument specifier of `\__deriv_scale_big:nnnn` when it was used (it was used with `:nnnm`).

v0.98
2020-07-20
Beta

- Fixed a bug related to the subscript argument caused by recent changes to the xparse package dated 2020-05-14 (the fix also works with earlier versions of xparse).

v1.0
2021-05-25

- Added new options for derivatives.
- Added new values for existing options.
- Added new commands for writing differentials.
- Changed the usage of `style-inf`.
- Changed the order argument to an option argument.
- Adjusted default values for certain options.
- Expanded math space keys to accept more general inputs.
- Changed `\derivset` to define default options.
- Merged the codes for ordinary and partial derivatives.
- Removed the mixed order argument.
- Removed the options `misc-add-delims` and `misc-remove-delims`.
- Replaced the commands in section 5 with new ones, as the old ones are deprecated.
- Code clean-up and optimization.
- Fixed code errors.

- Fixed documentation errors.
- Improved support for preventing options affecting nested derivatives/differentials.

v1.01
2021-05-28

- Changed the default value of the option `sep-end` due to issues.
- Fixed code errors due to changes introduced in version 1.0 to the option `style-inf`.
- Fixed documentation errors.

v1.1
2021-06-03

- Added package options to be used with `\usepackage[`⟨*options*⟩`]{derivative}`.
- Removed the hidden dependency on unicode-math when using X∃TEX or LuaTEX.
- Replaced `\symup` with `\mathrm` for simplicity.
- If mleftright is loaded, the default value of `scale-auto` is set to `mleftmright`.

v1.2
2022-07-09

- Code clean-up and optimisation that significantly speeds up usage of the package.
- Added new a sorting algorithm `sort-method=lexical`, which sorts terms alphabetically.
- Added the option `sort-lexical-reverse` to reverse the alphabetical sorting order.
- Added the option `switch-sort` to enable/disable the sorting algorithm for the mixed order.
- Fixed a bug related to `scale-`⟨*…*⟩`=none` when the corresponding delimiter option contained a period.
- Fixed documentation errors.

v1.3
2023-07-26

- Code clean-up and minor optimisation.
- Fixed documentation errors.
- Updated documentation to include more examples.
- Enhanced the algorithm that automatically calculates the mixed order to support fractional calculus.
- Introduced a new argument, `!`, for derivatives to enable shorthand notation.
- Added new options `style-inf-den`. and `style-inf-num` to set the infinitesimals in the denominator and numerator, respectively.
- Added new options `style-var-/`, `style-var-!` and `style-var-/!` to set the style of the derivatives.
- Added new options `scale-var-!` and `delims-var-!` for variable delimiters in shorthand mode.
- Added new options `scale-eval-!` and `delims-eval-!` for evaluation delimiters in shorthand mode.

v1.31
2023-11-04

- Fixed issue relating to the order when nesting derivatives (issues #14).