

mismath

Miscellaneous mathematical macros*

Antoine Missier
`antoine.missier@ac-toulouse.fr`

April 27, 2019

1 Introduction

This package provides some tools to get (automatically) mathematical constants e , i , π in upright shape and not in italic (sloping shape) like variables, according to the International Standards ISO 31-0:1992 to ISO 31-13:1992, superseded by ISO 80000-2:2009 (see [1] [2] [3] [4]).

Even if it is recommended to typeset vectors names in bold italic style [2] [3], they are often represented with arrows (particularly in scholar documents or in physics). To draw pretty arrows above vectors, we use `esvect` package by Eddie Sautrais [7] and we provide a few more macros related to vectors with arrows, in particular to improve the typesetting of the norm: $\|\overrightarrow{AB}\|$ instead of L^AT_EX version $\|\overrightarrow{AB}\|$ which is not adjusted, or worse $\|\overrightarrow{AB}\|$.

The package also provides other macros for:

- some standard operator names,
- a few useful aliases,
- improving some spacing in mathematical mode,
- systems of equations and matrices,
- `displaymath` layout in two columns.

To avoid incompatibility, a large majority of our macros will be defined only if there is not another command with the same name in the packages loaded before `mismath`. If a macro is already defined, compilation will produce a warning message and `mismath` definition will simply be ignored. To keep `mismath` command, either load `mismath` before the other package with which it is in conflict

*This document corresponds to `mismath` v1.2, dated 2019/04/27, thanks to François Bastouil for help in English translation.

for the name of the command (assuming the other package supports it), or use `\let\<command>\relax` before loading `mismath`.

`[\<options>]{mismath}`

The `amsmath` package is automatically loaded by `mismath` without option, and `mathtools` package also. For using `amsmath` with options (see [5]), these options can be added to `mismath`, or `amsmath` has to be loaded with the required options before `mismath`.

A recommendation, seldom observed, is to typeset uppercase Greek letters in italic shape like other variables [3]. This is automatically done with `fixmath` package by Walter Schmidt [9] or `isomath` by Günter Milde [10], but is not implemented in `mismath` because this rule is conflicting to the one used for instance in France where all mathematics capitals have to be typeset in upright shape¹. The choice of loading or not one of these packages remains thus to the user.

2 Usage

2.1 Mathematical constants

`\mathup` As for classic functions identifiers, *predefined* mathematical constants should be typeset in upright shape (generally in roman family), even if this practice is not really common and tedious to respect. To avoid to stuff a document with `\e` `\mathrm{e}` or `\mathrm{i}` (or better `\mathup{e}` and `\mathup{i}`²), `mismath` package provides `\e` command for the base of natural logarithm and `\i` or `\j` for imaginary numbers. One can notice that `\i` and `\j` already exist in L^AT_EX: using in LR mode, they produce “i” and “j” without the point so one can place accents on them, and in mathematical mode they produce “Latex warning: Command invalid in math mode”. Redefining `\i` and `\j` by `mismath` only concerns mathematical mode.

`\enumber` Nevertheless, it can be tiresome to type a lot of backslashes in a document
`\inumber` with many formulas with e or i. So a way is proposed here to free of it by placing
`\jnumber` `\enumber`, `\inumber` or `\jnumber` in the preamble: e, i or j will then automatically be set in upright shape in the whole document, no need to type `\e`, `\i` or `\j`. However, we can still get italicized *e*, *i* or *j* with L^AT_EX command `\mathit` or `\mathnormal`. Of course, this does not fully comply with L^AT_EX philosophy: in the document, objects have to be pointed out by their nature rather than their typographical characteristics, defined in the preamble. But these macros are really handy and thanks to them it is possible to put in the standards a document afterwards; moreover anyone is free to use them or not.

`\pinumber[\]`

Mathematical constant π should also be typeset in upright shape which differs from italicized π (see [3] and [4]). This recommendation is even less observed than the one concerning e and i [1]. The `upgreek` package by Walter Schmidt [8] makes it possible to typeset greek letters in upright font with commands such as `\upalpha`, `\upbeta`,... To avoid typing a lot of `\uppi`, `mismath`

¹The package `frenchmath` [12] takes this rule into account.

²`\mathup` is based on `\operatorfont` (from `amsopn` package, automatically loaded by `amsmath`). The `beamer` package uses sans serif as default math font, but `\mathrm` produces a font with serif in `beamer`. This problem is solved by using `\mathup` instead of `\mathrm`.

offers `\pinumber` macro, which has to be put in the preamble. This command loads the `upgreek` package with an optional `(font)` argument: `Symbol` (by default), `Euler` or `Symbolsmallscale` (see [8]). It also redefines `\pi` command to typeset all the `\pi` in the selected upright font.

By activating `\enummer`, `\inumber` and `\pinumber` in the preamble, you get the following example:

$$e^{i\pi} = -1 \qquad e^{i\pi} = -1$$

`\itpi` When `\pinumber` is activated, the original italic π can be produced with `\itpi`.

2.2 Vectors

`\vect` By default, `\vect` command³, produces vectors with arrows (thanks to `esvect` package by Eddie Sautrais⁴) which are much more elegant than those produced by L^AT_EX `\overrightarrow` command (giving \overrightarrow{AB}). The `esvect` package has an optional argument (one letter between `a` and `h`) defining the required type of arrow (see [7]). In `mismath`, `esvect` is loaded with the option `b`: `\vect{AB}` gives \overrightarrow{AB} . To choose another type of arrow, `esvect` must be called with the required option *before* `mismath`, for instance `\usepackage[d]{esvect}` will give the arrows produced by default in [7].

`\boldvect` With `\vect` it is also possible to typeset vector's names using bold italic, according to ISO recommendation [4], rather than arrows. For this, calling `\boldvect` will modify the behavior of `\vect`:

$$\begin{aligned} & \text{\texttt{[\boldvect \vect{v} \\ \text{\texttt{=}\lambda\vect{e}_x+\mu\vect{e}_y. \texttt{]}}}} \qquad \qquad \qquad \mathbf{v} = \lambda \mathbf{e}_x + \mu \mathbf{e}_y. \end{aligned}$$

`\boldvectcommand` By default `mismath` uses `\boldsymbol` command⁵ from `amssy` package, automatically loaded by `amsmath`. But other packages producing better bold italic can be preferred, for instance `\bm` from `bm` package or `\mathbf` from `fixmath` package or also `\mathbfbf` from `isomath` package. To achieve this, `\boldvectcommand` can be redefined: for instance `\renewcommand{\boldvectcommand}{\mathbf}`. By redefining `\boldvectcommand` to `\mathbf`, `\vect` produces vectors in bold *upright* style, which tends to be used instead of bold *italic* (but probably for bad reasons),

`\arrowvect` It is possible at any moment to get back to default behavior with inverse switch `\arrowvect`. These switches can be placed anywhere: inside mathematical mode or inside an environment (with local effect) or outside (with global effect).

`\hvect` When vectors with arrows are typeset side by side, the arrows can be set up a bit higher (with a phantom box containing *h*) to avoid inelegants effects:

- $\overrightarrow{AB} = \overrightarrow{u} + \overrightarrow{AC}$ is less than $\overrightarrow{AB} = \overrightarrow{u} + \overrightarrow{AC}$, produced by `\hvect{u}`;
- $\overrightarrow{a} \cdot \overrightarrow{b} = 0$ is less than $\overrightarrow{a} \cdot \overrightarrow{b} = 0$, produced by `\hvect{a}`.

³As for many macros of this package, the definition will take effect only if this macro is not defined before by another package.

⁴`esvect` provides `\vv` macro used in `\vect`.

⁵`\mathbf` gives upright bold style, even if used in combination with `\mathit`.

The `\boldvect` switch has no effect on `\hvect` macro which always typesets arrows on vectors (with `\vv` command from `esvect` package).

`\hvec` In a similar way, `\hvec` raises the little arrow produced by L^AT_EX command `\vec` (but only from height of *t* letter):

- $\mathcal{P} = \vec{f} \cdot \vec{v}$ is less than $\mathcal{P} = \vec{f} \cdot \vec{v}$, produced by `\hvec{v}`;
- $\vec{f} = m\vec{a}$ is less than $\vec{f} = m\vec{a}$, produced by `\hvec{a}`.

`\norm` The norm of a vector is classically produced with delimiters `\lVert` and `\rVert` (rather than `\|`) or `\left\Vert` and `\right\Vert` for delimiters adapting to the content. Unfortunately, these delimiters are always vertically centred, relatively to the middle of the base line, whereas vectors with arrows are asymetrics objects, the height above the middle of the base line being superior to the depth under it. The code `$_\norm{\vec{h}}$` raises the double bar to produce $\|\vec{h}\|$. Let's notice that the height of the bars don't adjust to content, but however to context: main text, indexes or exponents.

2.3 Standard operator names

`\di` The *differential* operator should be typeset in upright shape and not in italic, to make it different from variables (as mentioned in [1] [2] [3] [13]). To achieve this, `mismath` provides `\di` command⁶. See the following examples (notice thin space before the d, as for classic function's names):

$$\begin{aligned} & \int \int xy \, dx \, dy \\ & m \frac{d^2 x}{dt^2} + h \frac{dx}{dt} + kx = 0 \end{aligned}$$

This command can also stand for *distance*:

$$\lambda d(A, \mathcal{F}) + \mu d(B, \mathcal{H}).$$

`\P` In probability field, the proper use is to typeset capital letters P, E in upright shape to refer to probability and expectation⁷. To achieve this `mismath` provides `\P` and `\E`. Variance is normally denoted by Var (see later), but in some countries we can find V produced here by `\V`.

`\Par` `\P` command already existed to refer to the end of paragraph symbol ¶ and has been redefined, but this symbol can still be obtained with `\Par`.

`\probastyle` Some authors use also “blackbord bold” font to represent probability, expectation (and variance): $\mathbb{P}, \mathbb{E}, \mathbb{V}$. To get these “openwork” letters, use `\probastyle` macro to change appearance of `\P`, `\E` and `\V`: `\renewcommand\probastyle{\mathbb}`⁸.

⁶`\d` is already used in L^AT_EX to put a dot under a symbol and it seemed not a good idea to redefine this command.

⁷Upright shape must be used for any standard function identifier. L^AT_EX provides also `\Pr` which gives Pr.

⁸As for `\boldvect` and `\arrowvect`, effect is local to container environment.

`\mathbb` comes from `amsfonts` package (automatically loaded by `amssymb` but also available standalone) which has to be called in the preamble.

The following operator names are also defined in `mismath`:

<code>\adj</code>	adj	<code>\erf</code>	erf	<code>\rank</code>	rank
<code>\Aut</code>	Aut	<code>\grad</code>	$\overrightarrow{\text{grad}}$	<code>\Re</code>	Re
<code>\Conv</code>	Conv	<code>\id</code>	id	<code>\rot</code>	$\overrightarrow{\text{rot}}$
<code>\cov</code>	cov	<code>\Id</code>	Id	<code>\sgn</code>	sgn
<code>\Cov</code>	Cov	<code>\im</code>	im	<code>\spa</code>	span
<code>\curl</code>	$\overrightarrow{\text{curl}}$	<code>\Im</code>	Im	<code>\tr</code>	tr
<code>\divg</code>	div	<code>\lb</code>	lb	<code>\Var</code>	Var
<code>\End</code>	End	<code>\lcm</code>	lcm	<code>\Zu</code>	Z

By default, operators returning vectors, `\grad` and `\curl` (or its synonym `\rot` rather used in Europe), are written with an arrow on the top. When `\boldvect` is activated, they are typeset in bold font: **grad, curl, rot**. For covariance and identity function, `mismath` provides two notations, with or without a first capital letter, because they are both very frequent. On the other hand, `im` stands for the image of a linear transformation (like `ker` for the kernel) and `Im` is the imaginary part of a complex number.

`\oldRe` `\Re` and `\Im` macros already existed, to refer to real and imaginary part of a complex number, producing outdated symbols \Re and \Im . They have been redefined according to actual use, as mentionned in the above table, but it is still possible to get old symbols with `\oldRe` and `\oldIm`. Notice that `\div` and `\span` already exist and haven't been redefined, therefore `\divg` and `\spa` macros.

Some (inverse) circular or hyperbolic functions, missing in \LaTeX , are also provided by `mismath`:

<code>\arccot</code>	arccot	<code>\arsinh</code>	arsinh	<code>\arcoth</code>	arcoth
<code>\sech</code>	sech	<code>\arcosh</code>	arcosh	<code>\arsech</code>	arsech
<code>\csch</code>	csch	<code>\artanh</code>	artanh	<code>\arcsch</code>	arcsch

`\bigO` Asymptotic comparison operators (in Landau notation) are obtained with `\bigO` or `\bigo` and `\lito` commands:

$$n^2 + \mathcal{O}(n \log n) \quad \text{or} \quad n^2 + O(n \log n) \quad \text{and} \quad e^x = 1 + x + o(x^2).$$

2.4 A few useful aliases

In the tradition of Bourbaki and D. Knuth, proper use requires that classic sets of numbers are typeset in bold roman: **R, C, Z, N, Q**, openwork letters ($\mathbb{R}, \mathbb{Z}, \dots$) being restricted to writing at blackboard [13]; and likewise to designate a field: **F** or **K** (Körper in German). We get these symbols with the following macros:

`\R, \C, \Z, \N, \Q, \F, \K.`

2.5 Fixing some spacing in mathematical formulas

`\mul` Multiplication symbol obtained with `\times` produces the same spacing than addition or subtraction operators, whereas division obtained with `/` is tighten around its arguments. This actually hides the priority of the multiplication on `+` and `-`. This is why `mismath` provides `\mul` macro, behaving like `/` (ordinary symbol) and leaves less space around than `\times`:

$\lambda + \alpha \times b - \beta \times c$ is less than $\lambda + \alpha \times b - \beta \times c$, obtained with `\mul`.

`\then` `\then` macro produces the symbol \implies but leaving more space around it, as for
`\txt` standard macro `\iff` does it with \iff ; in a similar way, `\txt` uses `\text` macro from `amstext` package (automatically loaded by `amsmath`) leaving em quad spaces (`\quad`) around the text:

`\ln x=a` `\then x=e^a` `\txt{or}` `\ln x=a` `\Longrightarrow x=e^a`
 $\ln x = a \implies x = e^a \quad \text{or} \quad \ln x = a \implies x = e^a$

`\pow` When typesetting an exponent after a closing parenthesis produced by `\right)`, the exponent is little to far from the parenthesis. The command `\pow{<expr>}{<pow>}` sets `<expr>` between the parentheses and puts the exponent `<pow>` slightly closer to the parenthesis. Compare:

$$e^a \sim \left(1 + \frac{a}{n}\right)^n \quad \text{and} \quad e^a \sim \left(1 + \frac{a}{n}\right)^n.$$

`\abs` Absolute value (or modular for a complex number) should be typeset with `\lvert ... \rvert` rather than `|` which doesn't respect correct spaces for delimiters; for bars whose height has to adapt to content, we use `\left\lvert ... \right\rvert` or, more simply, `mismath` command `\abs{...}` which is equivalent¹⁰.

`\lfrac` This macro behaves like `\frac` but with medium spaces around the arguments, so the corresponding fraction bar seems to be a little bit longer:

$$\left[\frac{\lvert z \rvert}{\lvert \frac{z_1 - z_2}{z_1 + z_2} \rvert} \right]$$

`[]` Brackets symbols `[` and `]` have been redefined for mathematical mode because, in standard L^AT_EX, the space before them can be unsuitable¹¹:

$$x \in]0, \pi[\cup]2\pi, 3\pi[\quad \begin{cases} \text{without mismath} \\ \text{with mismath} \end{cases}$$

In our code, `[` and `]` symbols are not defined anymore as delimiters. One can regret it because a line break could occur between the two, but in addition to the fact that it works very well like that for spaces (because these delimiters are most of the time preceded or followed by relational, binary or punctuation symbols), it is always possible to transform them into delimiters with `\left` `\right`¹².

¹⁰Another solution is to define `\abs` with the `\DeclarePairedDelimiter` command from the `mathtool` package [6].

¹¹The `interval` package [11] gives another solution, less direct, based on an `\interval` macro.

¹²Is L^AT_EX standard definition of `[` as `\mathopen` really appropriate where this symbol could almost also logically have been defined as `\mathclose`?

2.6 Environments for systems and small matrices

system To produce a system of equations, `mismath` provides `system` environment:

$$\begin{array}{l} \text{\$}\backslash\text{begin}\{\text{system}\} \\ \quad x=1+2t \ \backslash\backslash \ y=2-t \ \backslash\backslash \ z=-3-t \\ \text{\$}\backslash\text{end}\{\text{system}\} \end{array} \quad \left\{ \begin{array}{l} x = 1 + 2t \\ y = 2 - t \\ z = -3 - t \end{array} \right.$$

`system`[(*coldef*)] By default, a system is written like an `array` environment with only one column, left aligned. The environment has an optional argument to create several columns, specifying their alignment, with the same syntax than `array` environment of L^AT_EX: `\begin{system}[cl]` produces a two-column system, the first one being centred, the second being left aligned, such as in the following example:

$$\begin{array}{l} \text{\$}\backslash\text{begin}\{\text{system}\}[cl] \\ \quad y \ \& \ = \text{\dfrac{1}{2}}x - 2 \ \backslash\backslash[1ex] \\ \quad (x,y) \ \& \ \neq (0,-2) \\ \text{\$}\backslash\text{end}\{\text{system}\} \end{array} \quad \left\{ \begin{array}{l} y = \frac{1}{2}x - 2 \\ (x,y) \neq (0,-2) \end{array} \right.$$

`\systemstretch` Default spacing between the lines of a `system` environment has been slightly enlarged compared to the one from `array` environments (from 1.2 factor). It is possible to change this spacing by typing `\renewcommand{\systemstretch}{\stretch}` inside the current mathematical environment (for a local change) or outside (for a global change). By default, `stretch`'s value is 1.2. It is also possible to use occasionally a carriage return with a spacing option such as it has been done above with `\backslash[1ex]`.

Another example with `\begin{system}[l@{\quad}l]`¹³:

$$\left\{ \begin{array}{ll} -5 + 3s = 8 + 2t & (r_1) \\ 1 + 2s = 3t & (r_2) \\ -2s = 8 + 2t & (r_3) \end{array} \right. \iff \left\{ \begin{array}{ll} -5 + 5s = 0 & (r_1 \leftarrow r_1 - r_3) \\ 1 = 8 + 5t & (r_2 \leftarrow r_2 + r_3) \\ 1 + 2s = 3t & (r_3 \leftarrow r_2) \end{array} \right.$$

`\systemsep` This last example could have also be produced with `cases` environment from `amsmath` package, although `cases` places mathematical expressions closer to the bracket (which is logical considering it's use). `\systemsep` macro enables to set the gap between the bracket and the expressions, set by default to `\medspace`. It is possible to reduce this gap, for instance: `\renewcommand{\systemsep}{\thinspace}` or to enlarge it with `\thickspace` (and with `\renewcommand{\systemsep}{}` we get back to what `cases` do).

spmatrix `amsmath` package provides various environments to typeset matrices: for instance `pmatrix` that automatically puts parenthesis around the matrix or `smallmatrix` that typesets a small matrix that can even be inserted on a text line. We provide a combination of the two with `spmatrix`:

$$\text{\$}\backslash\text{vec}\{u\}\backslash\text{begin}\{\text{spmatrix}\}-1\backslash\backslash 2\backslash\text{end}\{\text{spmatrix}\}\text{\$} \text{ yields } \vec{u} \begin{pmatrix} -1 \\ 2 \end{pmatrix}.$$

`mathtools` package enhance `amsmath` matrices environments and provides also a small matrix environment with parenthesis. Moreover, with starred version

¹³`@{...}` sets inter-column space.

`\begin{psmallmatrix*}[\langle col \rangle]`, you can choose the alignment inside the columns (c, l or r). But sadly, the space before the left parenthesis is too narrow regarding to the space inside the parenthesis. Compare previous $\vec{u}\left(\begin{smallmatrix} -1 \\ 2 \end{smallmatrix}\right)$ with $\vec{u}\left(\begin{smallmatrix} -1 \\ 2 \end{smallmatrix}\right)$.

2.7 Displaymath in double column

`\mathcols` `mathcols` environment activates mathematical mode and enables to compose “long” calculation on two columns, separated with a central line, as shown in the following example. But it is necessary to load `multicol` package in the preamble, action that `mismath` does not automatically do.

$$\begin{array}{l|l}
 \frac{1}{2 \times \left(\frac{1}{4}\right)^n + 1} \geq 0.999 & \iff 4^n \geq 1998 \\
 \iff 1 \geq 1.998 \left(\frac{1}{4}\right)^n + 0.999 & \iff n \ln 4 \geq \ln(1998) \\
 \iff 0.001 \geq \frac{1.998}{4^n} & \iff n \geq \frac{\ln(1998)}{\ln 4} \approx 5.4 \\
 & \iff n \geq 6
 \end{array}$$

`\changeacol` Alignment is produced using classic `&` and `\\` and `\changeacol` macro causes a change of column.

```

\begin{mathcols}
    & \frac{1}{2 \mul \pow{\frac{1}{4}}{n} + 1} \geq 0.999 \\
    \iff & 1 \geq 1.998 \pow{\frac{1}{4}}{n} + 0.999 \\
    \iff & 0.001 \geq \frac{1.998}{4^n} \\
\changeacol
    & \iff 4^n \geq 1998 \\
    & \iff n \ln 4 \geq \ln(1998) \\
    & \iff n \geq \frac{\ln(1998)}{\ln 4} \approx 5.4 \\
    & \iff n \geq 6
\end{mathcols}

```

3 Implementation

```

1 \DeclareOption*{\PassOptionsToPackage{\CurrentOption}{amsmath}}
2 \ProcessOptions \relax
3 \@ifpackageloaded{amsmath}{}{\RequirePackage{amsmath}}
4 \@ifpackageloaded{esvect}{}{\RequirePackage[b]{esvect}}
5 \RequirePackage{ifthen}
6 \RequirePackage{xspace}
7 \RequirePackage{mathtools}

```

The above conditional packages loading avoids “option clash” errors if the packages have been previously loaded with options.

The two following commands are meta commands for conditional macro definitions with warning message if the macro already exists.

```

8 \newcommand\@mwarning[1]{

```

```

9     \PackageWarning{mismath}{
10         Command \ #1 already exist and will not be redefined
11     } % TODO \bslash from doc.sty doesn't produce a blank space after \
12 }
13 \newcommand\@mmacro[2]{
14     \ifundefined{#1}{
15         \expandafter\def\csname #1\endcsname{#2}
16     }\@mwarning{#1}}
17 }
18 \newcommand\@moperator[3][\% this macro is ugly, TODO by default #1=#3
19 \ifthenelse{equal{#1}{}}{
20     \ifundefined{#3}{
21         \DeclareMathOperator{#2}{#3}
22     }\@mwarning{#3}}
23 }{
24     \ifundefined{#1}{
25         \DeclareMathOperator{#2}{#3}
26     }\@mwarning{#1}}
27 }
28 }
29
30 \DeclareSymbolFont{up}{\encodingdefault}{\familydefault}{m}{n}
31

```

To work correctly with beamer package, we did not use `\mathrm` but `\mathup` (based on `mathopn` `\operatorfont` to produce the correct upright shape font). This command works also fine with other sans serif fonts like `cmbright`.

`\AtBeginDocument` is necessary to redefine `\i` after calling `hyperref` package which overwrites the `\i` definition. Moreover for `beamer`, `\enumber` must use the `familydefault` font defined by this package (sans serif), therefore the `\AtBeginDocument` inside the macro (otherwise it has no effect). The same holds for `\inumber` and `\jnumber`.

```

32 \providecommand{\mathup}[1]{\operatorfont #1}
33 \@mmacro{e}{\mathup{e}}
34 \AtBeginDocument{\let\oldi\i \let\oldj\j
35     \renewcommand{\i}{\TextOrMath{\oldi}{\mathup{i}}}
36     \renewcommand{\j}{\TextOrMath{\oldj}{\mathup{j}}}
37 }
38 \newcommand{\enumber}{
39     \AtBeginDocument{\DeclareMathSymbol{e}\mathalpha{up}{‘e}}
40 }
41 \newcommand{\inumber}{
42     \AtBeginDocument{\DeclareMathSymbol{i}\mathalpha{up}{‘i}}
43 }
44 \newcommand{\jnumber}{
45     \AtBeginDocument{\DeclareMathSymbol{j}\mathalpha{up}{‘j}}
46 }
47 \newcommand*{\pinumber}[1][Symbol]{
48     \@ifpackageloaded{upgreek}{\usepackage[#1]{upgreek}}
49     \let\itpi\pi

```

```

50 \renewcommand{\pi}{\uppi}
51 }
52
53 \newboolean{arrowvect}
54 \setboolean{arrowvect}{true}
55 \newcommand{\arrowvect}{\setboolean{arrowvect}{true}}
56 \newcommand{\boldvect}{\setboolean{arrowvect}{false}}
57 \newcommand{\boldvectcommand}{\boldsymbol} % needs bm package
58 \@mmacro{vect}{\ifthenelse{\boolean{arrowvect}}{\vv}{\boldvectcommand}}
59 \newcommand*{\hvect}[1]{\vv{\vphantom{h}#1}}
60 \newcommand*{\hvec}[1]{\vec{\vphantom{t}#1}}
61
62 \newcommand*{\@norm}[1]{
63     \mbox{\raisebox{1.75pt}{\mathstrut\bigl\Vert$}} #1
64     \mbox{\raisebox{1.75pt}{\mathstrut\biggr\Vert$}} }
65 % works better than with relative length
66 \newcommand*{\@@norm}[1]{
67     \mbox{\footnotesize\raisebox{1pt}{\mathstrut\Vert$}} #1
68     \mbox{\footnotesize\raisebox{1pt}{\mathstrut\Vert$}} }
69 \newcommand*{\@@@norm}[1]{
70     \mbox{\tiny\raisebox{1pt}{\mathstrut\Vert$}} #1
71     \mbox{\tiny\raisebox{1pt}{\mathstrut\Vert$}} }
72 \providecommand*{\norm}[1]{
73     \mathchoice{\@norm{#1}}{\@norm{#1}}{\@@norm{#1}}{\@@@norm{#1}} }
74
75 \newcommand{\di}{\mathop{}\!\!\mathup{d}}
76 \newcommand\probastyle{}
77 \let\Par\P % end of paragraph symbol
78 \renewcommand{\P}{\operatorname{\probastyle{P}}}
79 \@mmacro{E}{\operatorname{\probastyle{E}}}
80 \@mmacro{V}{\operatorname{\probastyle{V}}}
81 \newcommand{\PEupright}{
82     \AtBeginDocument{% necessary for working with beamer
83         \DeclareMathSymbol{P}{\mathalpha{up}}{P}
84         \DeclareMathSymbol{E}{\mathalpha{up}}{E}
85     }
86 }
87
88 \@moperator{\adj}{adj}
89 \@moperator{\Aut}{Aut}
90 \@moperator{\Conv}{Conv}
91 \@moperator{\cov}{cov}
92 \@moperator{\Cov}{Cov}
93 \@mmacro{curl}{\operatorname{\vect{\mathup{curl}}}}
94 \@moperator{divg}{\divg}{div}
95 \@moperator{\End}{End}
96
97 \@moperator{\erf}{erf}
98 \@mmacro{grad}{\operatorname{\vect{\mathup{grad}}}}
99 \@moperator{id}{id} % mathop or mathord ?

```

```

100 \@operator{\Id}{Id}
101 \@operator{\im}{im}
102 \let\oldIm\Im \renewcommand{\Im}{\operatorname{Im}}
103 \@operator{\lb}{lb}
104 \@operator{\lcm}{lcm}
105
106 \@operator{\rank}{rank}
107 \let\oldRe\Re \renewcommand{\Re}{\operatorname{Re}}
108 \@mmacro{rot}{\operatorname{\vect{\mathup{rot}}}}
109 \@operator{\sgn}{sgn}
110 \@operator[spa]{\spa}{span}
111 \@operator{\tr}{tr}
112 \@operator{\Var}{Var}
113 \@operator[Zu]{\Zu}{Z}
114
115 \@operator{\arccot}{arccot}
116 \@operator{\sech}{sech}
117 \@operator{\csch}{csch}
118 \@operator{\arsinh}{arsinh}
119 \@operator{\arcosh}{arcosh}
120 \@operator{\artanh}{artanh}
121 \@operator{\arcoth}{arcoth}
122 \@operator{\arsech}{arsech}
123 \@operator{\arcsch}{arcsch}
124
125 \@operator[bigO]{\bigO}{\mathcal{O}}
126 \@operator[bigo]{\bigo}{\mathcal{O}}
127 \@operator[lito]{\lito}{\mathcal{O}}
128
129 \newcommand{\mathset}{\mathbf}
130 \@mmacro{R}{\ensuremath{\mathset{R}}\xspace}
131 \@mmacro{C}{\ensuremath{\mathset{C}}\xspace}
132 \@mmacro{N}{\ensuremath{\mathset{N}}\xspace}
133 \@mmacro{Z}{\ensuremath{\mathset{Z}}\xspace}
134 \@mmacro{Q}{\ensuremath{\mathset{Q}}\xspace}
135 \@mmacro{F}{\ensuremath{\mathset{F}}\xspace}
136 \@mmacro{K}{\ensuremath{\mathset{K}}\xspace}
137
138 \@mmacro{ds}{\displaystyle}
139 \@mmacro{dlim}{\lim\limits}
140 \@mmacro{dsum}{\sum\limits}
141 \@mmacro{dcup}{\bigcup\limits}
142 \@mmacro{dcap}{\bigcap\limits}
143 \@mmacro{lbar}{\overline}
144 \providecommand*{\hlbar}[1]{\overline{\vphantom{h}\#1}}
145 \@mmacro{eqdef}{\stackrel{\mathup{def}}{=}}
146 \@mmacro{unbr}{\underbrace}
147 \@mmacro{iif}{if and only if\xspace}
148
149 \@mmacro{mul}{\mathord{\times}}

```

```

150 \@macro{then}{\ \Longrightarrow \ \mbox{ } }
    Without \mbox{ } the space produced by \ would be suppressed in tabulars.

151 \providecommand*\txt[1]{\quad\text{#1}\quad}
152 \providecommand*\pow[2]{\left( #1 \right)^{\!#2}}
153 \providecommand*\abs[1]{\left\vert\!#1\right\vert}
154 \providecommand*\lfrac[2]{\frac{\!#1\!}{\!#2\!}}
155 \DeclareMathSymbol{}\mathord{up}{093} % originally \mathclose
156 \DeclareMathSymbol{}\mathord{up}{091} % originally \mathopen
157
158 \newcommand{\systemstretch}{1.2}
159 \newcommand{\systemsep}{\medspace}
160 \newenvironment{system}[1][1]{
161     \renewcommand{\arraystretch}{\systemstretch}
162     \setlength{\arraycolsep}{0.15em}
163     \left\{\begin{array}{@{\systemsep}#1@{}} %
164 }\end{array}\right.}
165
166 \newenvironment{spmatrix}{
167     \left(\begin{smallmatrix}
168 }\end{smallmatrix}\right)}
169
170 \newenvironment{mathcols}{% needs multicol package
171     \renewcommand{\columnseprule}{0.1pt}
172     \begin{multicols}{2}
173         \par\noindent\hfill
174         \begin{math}\begin{aligned}\displaystyle
175 }{%
176         \end{aligned}\end{math} \hfill\mbox{ }
177     \end{multicols}
178 }
179 \newcommand{\changepcol}{%
180     \end{aligned}\end{math} \hfill\mbox{ }
181     \par\noindent\hfill
182     \begin{math}\begin{aligned}\displaystyle
183 }

```

References

- [1] *Typesetting mathematics for science and technology according to ISO 31/XI*, Claudio Beccari, TUGboat Volume 18 (1997), No. 1.
- [2] *Typefaces for Symbols in Scientific Manuscripts*.
<https://www.physics.nist.gov/cuu/pdf/typefaces.pdf>.
- [3] *On the Use of Italic and up Fonts for Symbols in Scientific Text*, I.M. Mills and W.V. Metanomski, ICTNS (Interdivisional Committee on Nomenclature and Symbols), dec 1999.
- [4] *ISO 80000-2*. https://en.wikipedia.org/wiki/ISO_80000-2

- [5] *The `amsmath` package*. Frank Mittelbach, Rainer Schöpf, Michael Downes, Davis M. Jones, David Carlisle, CTAN, v2.17b 2018/12/01.
- [6] *The `mathtool` package*. Morten Høgholm, Lars Madsen, CTAN, v1.21 2018/01/08.
- [7] *Typesetting vectors with beautiful arrow with $\text{\LaTeX} 2_{\epsilon}$* . `esvect` package by Eddie Sautrais, CTAN, v1.3 2013/07/11.
- [8] *The `upgreek` package for $\text{\LaTeX} 2_{\epsilon}$* , Walter Schmidt, CTAN, v2.0 2003/02/12.
- [9] *The `fixmath` package for $\text{\LaTeX} 2_{\epsilon}$* , Walter Schmidt, CTAN, v0.9 2000/04/11.
- [10] *isomath. Mathematical style for science and technology*. Günter Milde, CTAN, v0.6.1 04/06/2012.
- [11] *The `interval` package*. Lars Madsen, CTAN, v0.3 2014/08/04.
- [12] *L'extension `frenchmath`*. Antoine Missier, CTAN, v1.2 2019/04/27.
- [13] *The Not So Short Introduction to $\text{\LaTeX} 2_{\epsilon}$* . `lshort` package by Tobias Oetiker, Hubert Partl, Irene Hyna and Elisabeth Schlegl, CTAN, v6.2 2018/02/28.
- [14] *The \LaTeX Companion*. Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, Chris Rowley, 2nd edition, Pearson Education, 2004.