Typesetting Mathematics in LaTeX

Petra Harwin

October 9, 2012

Contents

1 $\mathcal{AMS}$-LaTeX

The American Mathematical Society (AMS) have produced an extension for LaTeX called $\mathcal{AMS}$-LaTeX that makes it easier to typeset difficult mathematics in LaTeX. To use it we must use the package `amsmath`. To use a package you must include the following line in the preamble of your LaTeX document

```
\usepackage[options]{packagename}
```

with options being any particular options you require for the package. (I don't use any for `amsmath`.)

For the purposes of general mathematical typesetting I would suggest that you extend `amsmath` by loading one other $\mathcal{AMS}$ package:

```
\usepackage{amsmath,amssymb}.
```

1.1 Using \usepackage

I can load N packages at once by using the format

```
\usepackage{package1,package2,...,packageN},
```

which is equivalent to

```
\usepackage{package1}
\usepackage{package2}
⋮
\usepackage{packageN}.
```

However, if you need individual options for each package then you must use the second type of formatting.

1.2 Descriptions

`amsmath` This package is the primary enhancement package of $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX. $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX provides additional mathematical environments for multiline equations, more symbols in boldface type, easier construction of new symbols and much more. `amsmath` automatically loads `amsgen`, `amsbsy`, `amsopn` and `amstext`.

`amssymb` This package makes many mathematical symbols available that are not available in plain LaTeX. Look at the Tables on pages 552–554 of Kopka & Daly or §8.9 in Mittlebach & Goossens for a full list. `amssymb` automatically loads `amsfonts`.

2 Single-line formulæ

Mathematical formulæ are generally typeset in one of two ways: *inline*, meaning as part of the current line, or *displayed*, meaning on a separate line. For now we will concentrate on single-line formulæ.

This $u(x,t) = \int_{\mathbb{R}^N} f((x-z)t^{-1/2})u_0(z)\,\mathrm{d}z$ is an inline formula and this

$$u(x,t) = \int_{\mathbb{R}^N} f((x-z)t^{-1/2})u_0(z)\,\mathrm{d}z \tag{1}$$

is a displayed formula. Notice the difference in typesetting.

The first formula is generated by typing the mathematical content (we'll get onto this next) inside a pair of dollar signs: `$mathematics$`, whilst the second formula uses the `equation` environment:

```
\begin{equation}
mathematics
\end{equation}.
```

(To suppress the numbering in any environment use `\begin{envname*} mathematics \end{envname*}`.)

3 Basic mathematics

When typing mathematics into LaTeX you must be inside a mathematical environment of some kind. There are many of these but for now we only know about inline formulæ and the equation environment. The basic operations are typed as you would expect and LaTeX inserts the appropriate space where needed. E.g. `$x+y$` produces $x+y$. Fractions can be achieved in displayed formulæ by using the `\frac{numerator}{denominator}` command:

$$\frac{a+b}{c} = \frac{a}{c} + \frac{b}{c}.$$

You could of course put this equation inline $\frac{a+b}{c} = \frac{a}{c} + \frac{b}{c}$ but the fractions look small and perhaps reverting to $(a+b)/c = a/c + b/c$ is better for inline formulæ.

3.1 Mathematical commands

Subscripts are generated by using an underscore _ and superscripts by the caret symbol ^. For example x^y is gained by typing `$x^y$`. Only the letter or number immediately after ^ is taken to be a superscript. Thus, to gain x^{y+z} one must type `$x^{y+z}$` so that LaTeX treats $y+z$ as a single object. The same rules apply to subscripts.

Greek letters are easily generated by typing `\greeklettername` for a lowercase letter or `\Greeklettername` for an uppercase letter. For example `$\delta$` produces δ while `$\Delta$` produces Δ.

A full list of available mathematical commands can be found Chapter 5 of Kopka & Daly or Chapter 8 of Mittlebach & Goossens.

3.2 Conventions

There are a few important conventions in mathematical typesetting that LaTeX does not automatically deal with. The first one is the typesetting of the "dt" at the end of the integral. Contrast the following two examples:

$$\int tdt, \quad \int t\,\mathrm{d}t.$$

The first one is what results if you just type `$\int t dt$`. The problem is that LaTeX ignores all white space inside mathematical environments, so tdt just ends up looking like lots of variables clumped together. To fix this you need to tell LaTeX that you want a small white space between the t and the dt. This is done by using the `\,` command. To make things clearer still it is conventional to type the d of dt in a Roman font. Thus, the integral should be produced by typing `$\int t \, \mathrm{d}t$`.

Also the complex number i and Euler's constant e are conventionally typed in Roman.

3.3 Operators

There are many mathematical operators built into LaTeX and you should be aware of these. For example, you may think that $\sin x$ is produced by typing `$sin x$` but as I have already explained, this will simply generate $sinx$. To typeset this properly you must use the LaTeX command `\sin`. In doing this it is important to remember to leave a white space at the end of the `\sin` command because typing `$\sinx$` rather than `$\sin x$` will result in an error. LaTeX will think that you are asking it for a command called `\sinx`. This is true of all latex commands - leave a white space after them.

Some operators have options. For example `\int` and `\lim`. Operator options are typed using the superscript and subscript commands. For example `$\int^a_b$` produces $\int^a_b$.

4 Numbering and referencing

LaTeX provides a very simple system for numbering and referencing equations. If we wish to refer to an equation we must simply insert a `\label` command as follows:

```
\begin{equation}
\label{soln}
u(x,t)=\int_{\mathbb{R}^N}
   f((x-z)t^{-1/2})u_0(z) \, \mathrm{d}z.
\end{equation}
```

Then we may refer to this equation with the command `\eqref{soln}`, which produces (1) since this was the equation I typed in section 2. The command `\eqref` is preferable to the command `\ref` here since it automatically includes the surrounding brackets and uses the correct fonts regardless of the type of font you are using. (`\ref` would simply produce 1.)

4.1 Customising equation numbering

You can tell LaTeX how you want your equations numbered by using the `\numberwithin` command in the preamble. For example, if an article is to have the equations in each section numbered as `sectionnumber.equationnumber` use

`\numberwithin{equation}{section}`

to redefine the equation numbers to include the section number.

The general syntax is `\numberwithin{ctr}{in_ctr}` and it defines `ctr` to be a subcounter of `in_ctr`, meaning that `ctr` is reset every time `in_ctr` is incremented. Here are a few examples
`\numberwithin{figure}{subsection}`
`\numberwithin{table}{section}`.

5 Long equations

We have dealt with single line displayed mathematics but what if you have one equation that is simply too long for one line? Thankfully, $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX copes very well with this if you use its `split` or `multline` environments.

5.1 The `multline` environment

The `multline` environment is a variant of the equation environment for a *single* formula that is too long for one line. Line breaks occur where the user forces them with the `\\`

command. By default the first line is left justified, the last line right justified and the lines in between are centred. The equation number will appear at the right of the last line.

$$\begin{multline}
(x+y)^n=x^n+nx^{n-1}y\\+\frac{n(n-1)}{1\cdot2}x^{n-2}y^2\\
+\frac{n(n-1)(n-2)}{1\cdot2\cdot3}x^{n-3}y^3+\ldots\\
+nxy^{n-1}+y^n \tag{2}
\end{multline}$$

was produced with the code

```
\begin{multline}
\label{2long}
(x+y)^n=x^n+nx^{n-1}y\\+\frac{n(n-1)}{1\cdot2}x^{n-2}y^2\\
+\frac{n(n-1)(n-2)}{1\cdot2\cdot3}x^{n-3}y^3+\ldots\\
+nxy^{n-1}+y^n
\end{multline}
```

To shift individual lines fully to the left or right use the commands `\shoveleft{line}` and `\shoveright{line}`. The entire formula text for that line except the `\\` is placed within their arguments.

$$\begin{multline}
(x+y)^n=x^n+nx^{n-1}y\\
\shoveright{+\frac{n(n-1)}{1\cdot2}x^{n-2}y^2}\\
\shoveleft{+\frac{n(n-1)(n-2)}{1\cdot2\cdot3}x^{n-3}y^3+\ldots}\\
+nxy^{n-1}=y^n \tag{3}
\end{multline}$$

was produced by the code

```
\begin{multline}\label{2long2}(x+y)^n=x^n+nx^{n-1}y\\
\shoveright{+\frac{n(n-1)}{1\cdot2}x^{n-2}y^2}\\
\shoveleft{+\frac{n(n-1)(n-2)}{1\cdot2\cdot3}x^{n-3}y^3
+\ldots}\\
+nxy^{n-1}=y^n
\end{multline}
```

5.2 The `split` environment

Like `multline`, the `split` environment is meant for a single equation that does not fit on one line. The difference is that in each line there is an alignment marker `&` such that the lines are horizontally positioned to line up the markers.

The `split` environment does not automatically change into math mode or produce an equation number. That is because it is designed to be used inside an math environment. The most basic syntax is

```
\begin{mathenv}
\label{equation label}
\begin{split}
maths & maths\\
& maths
\end{split}
\end{mathenv}
```

For example, the equation

$$\begin{split}\psi_k(\xi_0) =-c_k&\frac{(m-3)a}{(m-1)\xi_0}\\ &+\frac{\xi_0(\lambda_k-\mu_0)a}{2m|a|^{m-1}} -\frac{2a}{\xi_0},\end{split} \tag{4}$$

was generated by using the `split` environment. Notice that the equation number appears between the two lines, not at the end of the last line. The code used to generate this was

```
\begin{equation}
\begin{split}
\psi_k(\xi_0) =-c_k&\frac{(m-3)a}{(m-1)\xi_0}\\
&+\frac{\xi_0(\lambda_k-\mu_0)a}{2m|a|^{m-1}}
-\frac{2a}{\xi_0},
\end{split}
\end{equation}
```

6 The multiline environments `gather` and `align`

Now we know a little bit about typesetting mathematics in LaTeX we should look at some ways of displaying mathematical text on more than one line.

`gather` The `gather` environment centres all the lines of mathematical text within it. Lines are separated by \\.

`align` The `align` environment can line up entities by using the markers `&`. Lines are separated in the same way as in the `gather` environment.

Let's look at some examples.

6.1 An example using the gather environment

LaTeX automatically centres single lines of mathematics so `gather` is not needed for single-line mathematics. For multiline mathematics `gather` gives

$$\int g(C_1\psi_1 + C_2\psi_2 + C_3\psi_3) = 0, \tag{5}$$

$$C_1^2 + C_2^2 + C_3^2 = 1. \tag{6}$$

This was produced using the code:

```
\begin{gather}
\int g(C_1 \psi_1 + C_2 \psi_2 + C_3 \psi_3)=0,\\
C_1^2+C_2^2 + C_3^2= 1.
\end{gather}
```

6.2 Numbering in multiline environments

$\mathcal{AMS}$-LaTeX automatically numbers every line of a multiple-line display. This is not always what you want so the command `\notag` is available to suppress the numbering of a line. The command should be placed at the end of the line but before any \\ command. So, to number only the first line in our display we use the code

```
\begin{gather}
\int g(C_1 \psi_1 + C_2 \psi_2 + C_3 \psi_3)=0,\\
C_1^2+C_2^2 + C_3^2= 1.\notag
\end{gather}
```

This produces

$$\int g(C_1\psi_1 + C_2\psi_2 + C_3\psi_3) = 0, \tag{7}$$

$$C_1^2 + C_2^2 + C_3^2 = 1.$$

6.3 The align environment

The `align` environment is intended for multiple equations with horizontal alignment. Each line is split into aligned column pairs. Within each pair the first column is right justified against an ampersand and the second column is left justified against that ampersand. An ampersand is also used to separate the column pairs.

If each line consists of n column pairs then the number of ampersands per line will be $2n-1$: one ampersand for alignment within each column pair (giving n) and $n-1$ ampersands to separate the columns. The schematic below illustrates the structure for two lines of mathematics and three column pairs.

right&left & right&left & right&left\\
right&left & right&left & right&left

The green ampersands are used for alignment while the red ampersands separate column pairs.

Here is a simple use of align (1 column pair)

$$\begin{aligned}\int x(x-1)\,\mathrm{d}x&=\int x^2-x\,\mathrm{d}x\\&=\frac{x^3}{3}-\frac{x^2}{2}+C,\end{aligned}$$

which was generated by using the code

```
\begin{align*}
\int x(x-1)\,\mathrm{d}x&=\int x^2-x\,\mathrm{d}x\\
&=\frac{x^3}{3}-\frac{x^2}{2}+C,
\end{align*}
```

(Note the use of `align*` to automatically suppress all numbering.)

Here is a more complicated example using the `align` environment (3 column pairs).

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}x}x^n&=nx^{n-1}&\frac{\mathrm{d}}{\mathrm{d}x}\mathrm{e}^x&=\mathrm{e}^x&\frac{\mathrm{d}}{\mathrm{d}x}\sin x&=\cos x\\\frac{\mathrm{d}}{\mathrm{d}x}\frac{1}{x^n}&=-\frac{n}{x^{n+1}}&\frac{\mathrm{d}}{\mathrm{d}x}a^x&=a^x\ln a&\frac{\mathrm{d}}{\mathrm{d}x}\cos x&=-\sin x\end{aligned}$$

This was generated using the code

```
\begin{align*}
\frac{\mathrm{d}}{\mathrm{d}x}x^n&=nx^{n-1}&
\frac{\mathrm{d}}{\mathrm{d}x}\mathrm{e}^x&=\mathrm{e}^x&
\frac{\mathrm{d}}{\mathrm{d}x}\sin x&=\cos x\\
\frac{\mathrm{d}}{\mathrm{d}x}\frac{1}{x^n}&=-\frac{n}{x^{n+1}}&
\frac{\mathrm{d}}{\mathrm{d}x}a^x&=a^x\ln a&
\frac{\mathrm{d}}{\mathrm{d}x} \cos x &=-\sin x
\end{align*}
```

6.3.1 Variations of the `align` environment

$\mathcal{AMS}$-LaTeX provides two variations on the `align` environment: `flalign` and `alignat`. The first has exactly the same syntax as `align` but it inserts enough space between the column pairs to ensure that the whole line is filled:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}x}x^n&=nx^{n-1}&\frac{\mathrm{d}}{\mathrm{d}x}\mathrm{e}^x&=\mathrm{e}^x\\\frac{\mathrm{d}}{\mathrm{d}x}\frac{1}{x^n}&=-\frac{n}{x^{n+1}}&\frac{\mathrm{d}}{\mathrm{d}x}a^x&=a^x\ln a.\end{aligned}$$

The `alignat` environment does the opposite: no spacing is inserted automatically between the column pairs. It also takes the number of column pairs as a mandatory argument. The following has been generated using `\begin{alignat}{2} maths \end{alignat}`.

$$\begin{alignat}{2}
\frac{\mathrm{d}}{\mathrm{d}x}x^n &= nx^{n-1} &\quad \frac{\mathrm{d}}{\mathrm{d}x}\mathrm{e}^x &= \mathrm{e}^x\\
\frac{\mathrm{d}}{\mathrm{d}x}\frac{1}{x^n} &= -\frac{n}{x^{n+1}} & \frac{\mathrm{d}}{\mathrm{d}x}a^x &= a^x\ln a.
\end{alignat}$$

6.4 Nested alignment environments

The two environments `aligned` and `gathered` are available for creating alignment within part of an equation. As such they are designed to be placed inside an mathematics environment and do not automatically switch to math mode. Their contents and behaviour are otherwise the same as `align` and `gather`.

Both of these arguments take an optional argument `pos`:

`\begin{aligned}[pos] lines \end{aligned}`
`\begin{gathered}[pos] lines \end{gathered}`,

which takes the value `t` or `b` (top or bottom) to determine the vertical alignment when they appear beside other elements. `t` makes the top of the contents aligned vertically with the external baseline and `b` makes the bottom of the contents aligned vertically with the external baseline. (The default is that the contents are centred vertically around the external baseline.)

Below is a simple example showing how these environments work.

$$\begin{aligned} \alpha &=1\\ \beta &=2 \\ \gamma &=3
\end{aligned}
\quad\text{and}\quad
\begin{aligned}[t]\delta &=4\\ \eta &=5\\ \phi &=6
\end{aligned}
\quad\text{and}\quad
\begin{gathered}[b] s=x+y\\ d=u-v\\p=x\circ y
\end{gathered}$$

was generated using the code

```
\begin{equation*}
\begin{aligned} \alpha &=1\\ \beta &=2 \\ \gamma &=3
\end{aligned}
\quad\text{and}\quad
\begin{aligned}[t]\delta &=4\\ \eta &=5\\ \phi &=6
\end{aligned}
\quad\text{and}\quad
\begin{gathered}[b] s=x+y\\ d=u-v\\p=x\circ y
\end{gathered}
\end{equation*}
```

7 Putting Text in Displayed Mathematics

Often we wish to include some text inside our mathematics. This can be achieved simply by using `\text{text to include}` but a word should be said about spacing. LaTeX ignores all white space in math mode so you must create the appropriate amount of space yourself. LaTeX has various commands for this:

Command name	Short form	Demo	Command name	Short form	Demo
`\negthinspace`	`\!`	$\rfloor\!\lfloor$	`\thinspace`	`\,`	$\rfloor\,\lfloor$
`\negmedspace`		$\rfloor\negmedspace\lfloor$	`\medspace`	`\:`	$\rfloor\:\lfloor$
`\negthickspace`		$\rfloor\negthickspace\lfloor$	`\thickspace`	`\;`	$\rfloor\;\lfloor$
			`\quad`		$\rfloor\quad\lfloor$
			`\qquad`		$\rfloor\qquad\lfloor$

It also has the manual spacing command `\mspace{mu}`, which inserts space in the spacing units 'mu' (=1/18 em). E.g., `\mspace{-9mu}` puts in negative space of 1/2 em. Choose your spacing and keep it consistent.

Here are two very simple examples of text and spacing inside mathematics environments.

```
\begin{equation*}
\frac{1}{n}\to \infty \quad \text{as} \quad n\to 0
\end{equation*}
```

gives

$$\frac{1}{n}\to \infty \quad \text{as} \quad n\to 0.$$

```
\begin{align*}
&y=0 \qquad \text{when} \qquad x=6 \\
&y=7\qquad \text{when}\qquad x=2
\end{align*}
```

$$\begin{aligned}
&y=0 \qquad \text{when} \qquad x=6 \\
&y=7\qquad \text{when}\qquad x=2
\end{aligned}$$

7.1 `intertext`

Another command for inserting normal text within a displayed equation is `\intertext{text}`, which produces a left-justified line of text between the lines of the formula without affecting the alignment of the formula. It is useful for producing things like

$$(x+\mathrm{i}y)(x-\mathrm{i}y)=x^2+y^2 \tag{8}$$

while on the other hand

$$(x+\mathrm{i}y)^2 = x^2 + 2\mathrm{i}xy - y^2. \tag{9}$$

```
\begin{align}
(x+\mathrm{i}y)(x-\mathrm{i}y)&=x^2+y^2\\
\intertext{while on the other hand}
(x+\mathrm{i}y)^2&=x^2+2\mathrm{i}xy-y^2.
\end{align}
```

The alignment is preserved whereas it would be lost if we had two separate equation environments.

8 Bracket sizing

Mathematics often contains bracketing symbols, usually in pairs that enclose a formula. When printed these bracket symbols should be the same size as the formula they enclose. LaTeX provides a pair of commands `\left` and `\right` to accomplish this. They must be used together and the syntax for them is

`\left` *lbracket* formula `\right` *rbracket*

Thus, to gain

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{x^2}\right)=-\frac{2}{x^3}$$

use the code

```
\begin{equation*}
\frac{\mathrm{d}}{\mathrm{d}x}
\left(\frac{1}{x^2}\right)=-\frac{2}{x^3}
\end{equation*}
```

8.1 Ghost brackets

The `\left` and `\right` commands can only be used on a single line of a multiline formula. To get around this LaTeX provides a ghost bracket ".". This can be used in place of a bracket and prints nothing:

$$\begin{aligned}\frac{1}{2} = \biggl(&\frac{1}{8}+\frac{1}{8}\\ &+\frac{1}{4}\biggr)\end{aligned}$$

```
\begin{align*}
\frac{1}{2}=\left(
\frac{1}{8}+\frac{1}{8}\right.&\\
\left.+\frac{1}{4}\right)&
\end{align*}
```

Using a ghost bracket often results in cumbersome formatting since you may not put an alignment character between the `\left` and `\right` commands. To avoid this you can size your brackets manually.

8.2 Manual sizing of brackets

The following commands are available for manual bracket sizing:

`--`	$()\{\}[]$	`\big`	$\big(\big)\big\{\big\}\big[\big]$	`\Big`	$\Big(\Big)\Big\{\Big\}\Big[\Big]$
`\bigg`	$\bigg(\bigg)\bigg\{\bigg\}\bigg[\bigg]$	`\Bigg`	$\Bigg(\Bigg)\Bigg\{\Bigg\}\Bigg[\Bigg]$		

Our previous example can now be formatted in a more logical way

$$
\begin{aligned}
\frac{1}{2}=\bigg(\frac{1}{8}&+\frac{1}{8}\\
&+\frac{1}{4}\bigg)
\end{aligned}
$$

```
\begin{align*}
\frac{1}{2}=\bigg(\frac{1}{8}&+\frac{1}{8}\\
&+\frac{1}{4}\bigg)
\end{align*}
```

9 Subnumbering

Equations in multiline formulæ may be subnumbered, that is, the main equation number stays the same and a letter is appended to it. This is achieved by using the `subequations` environment. The `subequations` environment does not change into math mode so it must contain a maths environment. Its syntax is

```
\begin{subequations}
\label{mainlabel}
\begin{mathsenv}
maths
\end{mathsenv}
\end{subequations}
```

Note that there is a label before the maths environment begins. This is used to refer to the equation number without any appended letters. For example, the code

```
\begin{subequations}
\label{trig}
\begin{gather}
\sin^2(x)+\cos^2(x)=1\label{triga}\\
\sin(\alpha+\beta)=\sin(\alpha)\cos(\beta)
+\sin(\beta)\cos(\alpha)\label{trigb}\\
\cos(\alpha+\beta)=\cos(\alpha)\cos(\beta)
-\sin(\alpha)\sin(\beta)\label{trigc}
\end{gather}
\end{subequations}
```

produces

$$\sin^2(x)+\cos^2(x)=1 \tag{10a}$$

$$\sin(\alpha+\beta)=\sin(\alpha)\cos(\beta)+\sin(\beta)\cos(\alpha) \tag{10b}$$

$$\cos(\alpha+\beta)=\cos(\alpha)\cos(\beta)-\sin(\alpha)\sin(\beta) \tag{10c}$$

We can now refer to the whole group of equations (10) (using `\eqref{trig}`) or any subequation, e.g. (10a) (using `\eqref{triga}`).

10 The `cases` Environment

How would you go about producing a structure like this?

$$P_{r-j}=\begin{cases}0 & \text{if $r-j$ is odd}\\ r!\,(-1)^{(r-j)/2} & \text{if $r-j$ is even.}\end{cases} \tag{11}$$

$\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX has a nice environment to do this for you: `cases`. The `cases` environment does not automatically switch into math mode but is very easy to use. The structure (11) was produced using the following code

```
\begin{equation}
\label{cases}
P_{r-j}=\begin{cases}
0 &\text{if $r-j$ is odd}\\
r!\,(-1)^{(r-j)/2} &\text{if $r-j$ is even.}\end{cases}
\end{equation}
```

Note how the alignment characters work.

11 Matrices

Matrix environments can only be used inside maths environments. They all have the same syntax but the environment name changes depending on what type of brackets you wish to surround your matrix. The basic syntax uses & to separate columns and \\ to separate rows:

```
\begin{mathsenv}
\begin{matrix}
entry11 & entry12 & ... entry1n\\
entry21 & entry22 & ... entry2n\\
.
.
.
entrym1 & entrym2 & ... entrymn
\end{matrix}
\end{mathsenv}
```

For example, typing

```
\begin{equation*}
\begin{matrix}
1 & 0\\
0 & 1
\end{matrix}
\end{equation*}
```

will produce

$$\begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix}$$

Notice that the standard matrix environment produces no brackets around the entries. To rectify this we must learn about its variations.

11.1 The matrix environments

Here are the different types of matrix environment that can be used.

- `matrix` puts no brackets around the entries.
- `pmatrix` puts round brackets around the entries.
- `bmatrix` puts square brackets around the entries.
- `Bmatrix` puts curly brackets around the entries.
- `vmatrix` puts a vertical line either side of the entries.

- `Vmatrix` puts two vertical lines either side of the entries.

For example, typing

```
\begin{equation*}
A=
\begin{pmatrix}
1 & 0\\
0 & 1
\end{pmatrix}
\end{equation*}
```

will produce

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

11.2 Matrices in text

To produce a small matrix suitable for use in text, use the `smallmatrix` environment. This environment does not produce any brackets so these should be added manually. For example, the code

```
This is my small matrix: $\left(\begin{smallmatrix}
                                   1 & 0\\
                                   0 & 1
                                 \end{smallmatrix}\right)$
followed by enough text to show you that it doesn't increase
the space needed for a line.
```

produces
This is my small matrix: $\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ followed by enough text to show you that it doesn't increase the space needed for a line.