

Microsoft Word 2013 公式

高效使用攻略

Ver 1.0

王邦柱 苏鑫

2014 年 11 月 30 日

北京·石景山

目 录

前言	1
插入公式	2
空格	2
上下标	2
希腊字母	2
大型公式	2
分数	3
根式	3
运算符	4
积分	4
方程组	5
特殊括号	5
标音导数类符号	5
矩阵	6
分段函数	6
常用符号-输入法对照表	7

前言

近几天写学术报告，有大量的公式插入。使用 Microsoft Word 的公式插入，显示出来已经足够美观，但就是输入起来略感麻烦。特别是输入带大量角标的公式、大型矩阵和方程组的时候，点鼠标和对齐就成为非常令人头疼的事情，屡次有摔键盘的冲动。当然，我没有摔键盘，而是反思自己学艺不精，没能知晓 Microsoft Word 公式的诀窍而已。于是，专门拿出一定的时间去研究学习；于是，有了这个小册子。所谓“工欲善其事，必先利其器”，现在我输入公式的效率已经远非知道本文所列的这些技巧之前所能比。

效率的提高，意味着时间的节省，意味着可以有更多的时间去做其他有意义的事情。当然，也意味着省电，这是我们在力所能及的范围内为我国节能减排做出的贡献。

研究实践中，我和老婆不断互动交流讨论，我学到一个知识点/技巧后，她就开始提问，然后我去搜索解决。就这样，我们很快就涵盖了常用的公式快速输入方法。——提一句，这样的研习方法效率非常高，我们从开始查资料到完成这个小册子，一共就用了 6 个小时左右。——最后，一些不成系统的小点，她还给列了一个对照表。

作者是开源运动的支持和爱好者，非常愿意本小册子能在非盈利的模式下广泛传播，帮助更多的有需要的人提高公式输入效率，也欢迎有朋友以此为基础的再创作。但是我也希望在传播转载过程中保留我们的名字，改编、商业出版的，希望能事前通知我们让我们知晓。

可通过下列方式与作者联系、交流、讨论：ibondrew@gmail.com（邮箱）、<http://weibo.com/1596149063/profile>（新浪微博）、768948104@qq.com（支付宝）。

感谢您的阅读。

作者谨识
2014 年 11 月 30 日

插入公式

鼠标方法：插入→公式

快捷键：Alt +=（Alt 和=同时摁）Microsoft 的“线性公式”

所谓线性，是微软提出的一种使用 Unicode 编码数学公式的方法，它采用了部分(La)TeX 的记法，但对(La)TeX 的非“所见即所得”做了更人性化的处理。线性公式的输入即能让人看懂其数学含义且无歧义，输入后，内置引擎会以预定义的规则对输入进行解释，显示为专业数学形式。而对于字体、颜色等更高层级的设置，公式交给 Word 进行统一处理。

空格

公式中很少需要空格的输入，引擎会自动进行空格的处理，保证显示的比例最优。在公式应用中，空格键的主要作用是确认线性输入完成，告诉引擎进行“翻译”，以专业型显示公式。“翻译”结束后，空格会自动被删除。

上下标

上下标是最常用的数学输入。上标使用^，下标使用_。只要没有遇到运算符或者空格，^/_后面的输入都被认为是上下标内容。如果上/下标中有运算，请使用括号，但是引擎在显示的时候会把括号去掉，这体现了线性公式的特点：输入无歧义，显示符合数学规则。比如，输入 a^wert_asdf，得到 a_{asdf}^{wert} ；a^(b+c)_(d-e) 显示为 a_{d-e}^{b+c} 。左标同理，比如(_a^b)c，得到 ${}_a^b c$ 。上标的上标这种嵌套，直接按 a^b^c 形式输入，得到 a^{b^c} 。输入完先按一下空格，再按一下空格，看看发生什么，体会一下引擎是如何处理这种嵌套的。特别的，试试 a^b^c_d 会出现什么？

希腊字母

数学公式输入，希腊字母太常见。公式这里完全借鉴了 LaTeX，使用转义符 (\) 引导希腊字母的英文拼法来输入。比如\alpha，得到 α ；\Delta，得到 Δ 。即，小写首字母的单词得到小写希腊字母，大写首字母的得到大写希腊字母。配合上下标使用的例子：\mu_0^2 得到 μ_0^2 ，\rho_s 得到 ρ_s ，\omega^* 得到 ω^* 。

大型公式

大型公式指的是求和、求积等公式，命令分别是\sum、\prod 配合上下标使用，即可完成复杂大型公式的输入。 \sum_{k=1}^n k^2 得到 $\sum_{k=1}^n k^2$ 。这是大型公

式的行内公式的显示方法，即把上下标显示为右侧。行间公式会将上下标显示在上下，即这样：

$$\sum_{k=1}^n k^2$$

求极限也算是大型符号，不过它没有特殊符号，直接输入 `lim` 文本就行，引擎会认出来，自动进行上下标的调整。极限常用的右箭头，使用短线和右书名号配合而成 (\rightarrow)，引擎自动会解释为右箭头。无穷大符号输入方法为 `\infty`。例如 `lim_(x->\infty)(x-2)` 得到 $\lim_{x \rightarrow \infty} (x - 2)$ 。注意看，引擎确实认出了 `lim`，将 $x \rightarrow \infty$ 放到了它的下面。注意显示字体，`lim` 作为运算/函数名，没有倾斜显示。而 x 作为变量，做了倾斜显示。

分数

分数输入的时候，用的是手写中常用的 `/` 作为分数线。分子或分母中如果有运算符，注意使用括号进行界定。比如，`1+2/4` 的结果是 1.5 而不是 0.75。分式引擎的解释方法也是类似的，可以分别试试 `(a+b)/c` 和 `a+b/c`，得到的是 $\frac{a+b}{c}$ 和 $a + \frac{b}{c}$ ，如果就想使用斜线的形式显示分数线，那可以强制转义使用斜线，试试 `(a+b)\c`，得到 $(a + b)/c$ ，这在行间使用分式特别有用。此外，我们可以看到，使用横线作为分数线的时候，我们为了避免歧义，使用了括号，但是显示的时候，括号并不是必须的，所以引擎自动去掉了括号。而使用斜线作为分数线的时候，括号并没有被自动去掉。如果在横式中就想使用括号，那就使用两层括号，比如 `((a+b))/c`，会得到 $\frac{(a+b)}{c}$ 。

此外，英文的逗号、和句号会分别被解释为分位符和小数点，比如，`a/.3` 会被解释为 $\frac{a}{.3}$ ，`a/1,333` 会被解释为 $\frac{a}{1,333}$ 。

还有几个特殊的分数，就是微元/微分运算符。`\Delta y/\Delta x` 得到 $\frac{\Delta y}{\Delta x}$ ，小写的 `\delta y` 自然得到 $\frac{\delta y}{\delta x}$ 。偏微分符号叫做 `\partial`，所以 `\partial^2 y/\partial x^2` 得到 $\frac{\partial^2 y}{\partial x^2}$ 。

根式

最常见的二次根式，命令是 `\sqrt`，这和绝大多数程序设计语言中的说法是一样的。输入 `\sqrt` 后敲一个空格，然后输入根式内的表达式即可，同样，如果表达

式中有运算，一定要加括号。比如 $\sqrt[n]{a+n}$ 得到 $\sqrt[n]{a+n}$ ，而 $\sqrt[n]{a+n}$ 得到 $\sqrt[n]{a+n}$ 。多次根式怎么办？格式是这样的 $\sqrt[n]{a}$ ，其中 n 是次数， a 是被开方的数，比如 $\sqrt[n]{5a+s+v}$ 得到 $\sqrt[n]{5a+s+v}$ ，注意这里， $\&$ 作为分界符，已经明确它前面是次数，后面是开方内容，所以不用给 $a+s+v$ 加括号，也不会引起歧义。

运算符

加减乘除等简单运算符在键盘上都能找到。对于乘除，不同条件下，使用的符号还略有区别。以乘为例，有点乘和叉乘。 $a \cdot b$ 、 $a \times b$ 和 $a \div b$ 分别得到 $a \cdot b$ ， $a \times b$ ， $a \div b$ 。

不等号的命令是 \neq ，即 not equal 的意思。但是更直观的输入方法是 \neq ，自动被解释为 \neq 。同理的还有 \geq 、 \leq 、 \gg 、 \nlessgtr 被解释为 \geq 、 \leq 、 \gg 、 \nlessgtr 。试试 \neq 、 \neq ，看看得到什么。

另外，有一些符号只能用命令输入，不能在键盘上找到。常见符号如下表所示：

所见	输入
\exists	$\backslash exists$
\forall	$\backslash forall$
\in	$\backslash in$
\approx	$\backslash approx$
\equiv	$\backslash equiv$
\triangleq	$\backslash Deltaeq$
$\subset \subseteq$	$\backslash subset \quad \backslash subseteq$

它们的否定是运算符命令前加 \neq ，比如， \nexists 是 $\backslash exists$ 得到的。

为什么公式没有采用程序设计语言常用的 \neq 表示不等于？因为 \neq 在数学中还是阶乘的意思， $3 \neq 4$ 是解释成 $3 \neq 4$ ，还是 $3 \neq 4$ ？此外，还有一个案例，就是上文已经使用过的 \rightarrow 解释为右箭头的用法， \leftarrow 就不能解释为左箭头，因为还可能是 a 小于负 b ($a < -b$)。从这里，又能看出公式设计人性化的一面。

积分

积分也是一种大型运算符。命令是 \int ，比如 $\int_a^b 2\pi \sin x dx$ ，行内得到 $\int_a^b 2\pi \sin x dx$ ，行间得到

$$\int_a^b 2\pi \sin x dx$$

注意， \sin 没有被倾斜显示。公式引擎能认出绝大部分常用函数，将之作为

文本处理而不予以倾斜（倾斜的被认为是变量或者常量）。

多重积分，命令也特别简单，比如二重 \iint ，三重 \iiint 。圈积分就是 \oint ，二重圈积 \oiint 等等。

方程组

方程组的输入格式是 $\eqarray (eq1@eq2@eq3)$ ，其中的@是换行的意思。比如 $\eqarray (3x+4y=5@4x+5y=6)$ ，显示为

$$\begin{array}{l} 3x + 4y = 5 \\ 4x + 5y = 6 \end{array}$$

很多公式，我们需要对齐处理。比如上面的公式，如果把 4y 换成 400y，会显示为

$$\begin{array}{l} 3x + 400y = 5 \\ 4x + 5y = 6 \end{array}$$

很难看。公式借鉴了 LaTeX 的用法，使用&作为对齐点。比如 $\eqarray (3x+400y&=5@4x+5y&=6)$ ，显示为

$$\begin{array}{l} 3x + 400y = 5 \\ 4x + 5y = 6 \end{array}$$

实现了按等号=对齐。

类比一下，求和和求积中，有时候上下标不止一行。多行的话，使用 \eqarray 即可。比如 $\sum_{0 \leq i \leq m} \prod_{a < j < n} P(i,j)$ 得到

$$\sum_{\substack{0 \leq i \leq m \\ a < j < n}} P(i,j)$$

如果你显示的不对，注意空格的使用。

特殊括号

量子力学中，常用狄拉克符号，左侧的叫 $\bra{}$ ，右侧的叫 $\ket{}$ ，随便输一个 $\bra{\psi} \ket{\psi}$ ，得到 $\langle \psi | \psi \rangle$ ，中间的直线就是键盘上的直线|。这个直线同时也可以用来输入绝对值哦，输入绝对值的时候更要注意使用括号予以定界哦，不然绝对值符号容易配对配错哦。

模值是双竖线，不能用两根竖线代替。模值的输入方法是 $\norm{}$ ，比如 $\norm{Ax-b}$ ，得到 $\|Ax - b\|$ 。

标音导数类符号

声调、导数、矢量等符号是标在“正头顶”上的。输入方法是先输入表达式，然后输入相应的命令。比如 $a\dot{}$ 得到 \dot{a} ，按空格的时候慢着点，观察引擎是怎么解释的（按两下）。 $a\hat{}$ 、 $a\vec{}$ 、 $a\bar{}$ 、 $a\check{}$ 、 $a\tilde{}$ 、 $a\ddot{}$ 分别得到 \hat{a} \vec{a} \bar{a} \check{a} \tilde{a} \ddot{a} 。

矩阵

矩阵的输入和方程组的输入是完全类似的，引导命令为`\matrix` 或`\pmatrix`，区别在于后者会用括号把矩阵括起来，而前者不会。以 p 为例，输入`\pmatrix (a&b@c&d)`，得到

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

矩阵是分行列的，每一个元素都需要对齐。输入一个更复杂的矩阵作为例子演示：

`\pmatrix (a^2+b_2&c&\alpha_3@ \vdots&\ddots&\vdots@q&w&r)`，得到

$$\begin{pmatrix} a^2 + b_2 & c & \alpha_3 \\ \vdots & \ddots & \vdots \\ q & w & r \end{pmatrix}$$

注意省略号的输入，`\vdots` 中 v 表示 *vertical*，`\ddots` 中的 d 表示 *diagonal*，横向省略号...用`\cdots` 表示（ c 是啥，你猜？）。

学会了这种输入方法后，会感觉特别有效率。如果输入一个矩阵，用鼠标点来点去，去填补一个个的空位，绝对是非常闹心的。小规模矩阵还好，大规模的矩阵那绝对会让你有摔电脑的冲动啊。特别是，如果你的矩阵特别大，通过一个空格扩充多个空格的形式增加矩阵行列数，对齐问题更是愁死人！

分段函数

$$|x| = \begin{cases} x, & \text{if } x > 0 \\ -x, & \text{if } x \leq 0 \end{cases}$$

上述公式的输入方法是：`|x|=\cases{&x &,"if" x>0@-&x &,"if" x\leq 0}`。`\cases` 可以产生一个自动适应的左大括号，适用于分段函数。为了美观，合理使用`&`进行对齐。

常用符号-输入法对照表

你想得到	你该输入	备注
$\int_a^b f(x)dx$	<code>\int_a^b f(x)dx</code>	几重积分就写几个 i，例如，二重积分 \iint ： <code>\iint</code> 。
$\oint_S f(x)dx$	<code>\oint_s f(x)dx</code>	
$\overbrace{x+\cdots+x}^{k \text{ times}}$ $\underbrace{x+y+z}_{>0}$	<code>\overbrace (x+...+x)^(k “times”)</code> <code>\underbrace (x+y+z)_(>0)</code>	注意 <code>\overbrace</code> 后面的空格。“ ”内的内容作为文本而不是变量显示，即不是斜体。
$\overrightarrow{a+b}$ $\underline{a+b}$	<code>(a+b)\above -></code> <code>(a+b)\below-></code>	在 $a+b$ 上/下加箭头，其中 $a+b$ 的字号和位置作为正文显示。
$\xrightarrow{a+b}$ $\xrightarrow[a+b]$	<code>->\above (a+b)</code> <code>->\below (a+b)</code>	注意和上面区别。 $A \above B$ 的含义是，在 A 的“头顶”加 B，A 为主，B 为次（略小）。
$\binom{n}{k}$	<code>(n\atop k)</code>	注意和上面区别。 $A \atop B$ ，A 和 B 只有位置上的区别，没有大小的区别。
$\boxed{E=mc^2}$	<code>\rect (E=mc^2)</code>	
$\angle A + \angle B = \angle C$	<code>\angle A +\angle B=\angle C</code>	
$\because \therefore$	<code>\because \therefore</code>	如果显示不正确，看看自己单词是不是拼错了。学好英语多重要！
$a \perp b$	<code>a\bot b</code>	几何中常用的垂直符号。
$a \parallel b$	<code>a\parallel b</code>	平行符号。
$A \cap B \quad A \cup B$	<code>A\cap B, A\cup B</code>	记法很形象，口向下像帽子，口向上像茶杯。
$\angle A = 20^\circ$	<code>\angle A = 20\degree</code>	
$A = \emptyset$	<code>\emptyset</code>	
$\uparrow \rightarrow \leftarrow$ \leftrightarrow	<code>\downarrow \uparrow</code> <code>\rightarrow \leftarrow</code> <code>\leftrightarrow</code>	命名很有规律，记住一个即可。
$\Downarrow \Rightarrow \Leftarrow$ \Leftrightarrow	<code>\Downarrow</code>	首字母大写，其他依据上一栏自行脑补。
$\hbar \omega$	<code>\hbar \omega</code>	量子力学常用的约化普朗克常量。

$\nabla\varphi = 0$	<code>\nabla \varphi =0</code>	注意 <code>\varphi</code> 和 <code>\phi</code> 的区别， φ vs ϕ
$a + b = c$	<code>\scripta +\scriptb=\scriptC</code>	字母手写花体就是在字母前加 <code>\script</code> ，与字母间无空格。
$F \propto v^2$	<code>F\propto v^2</code>	
$\vec{a} \vec{a}$	<code>a\vec{ } a\hvec{ }</code>	注意两者区别。