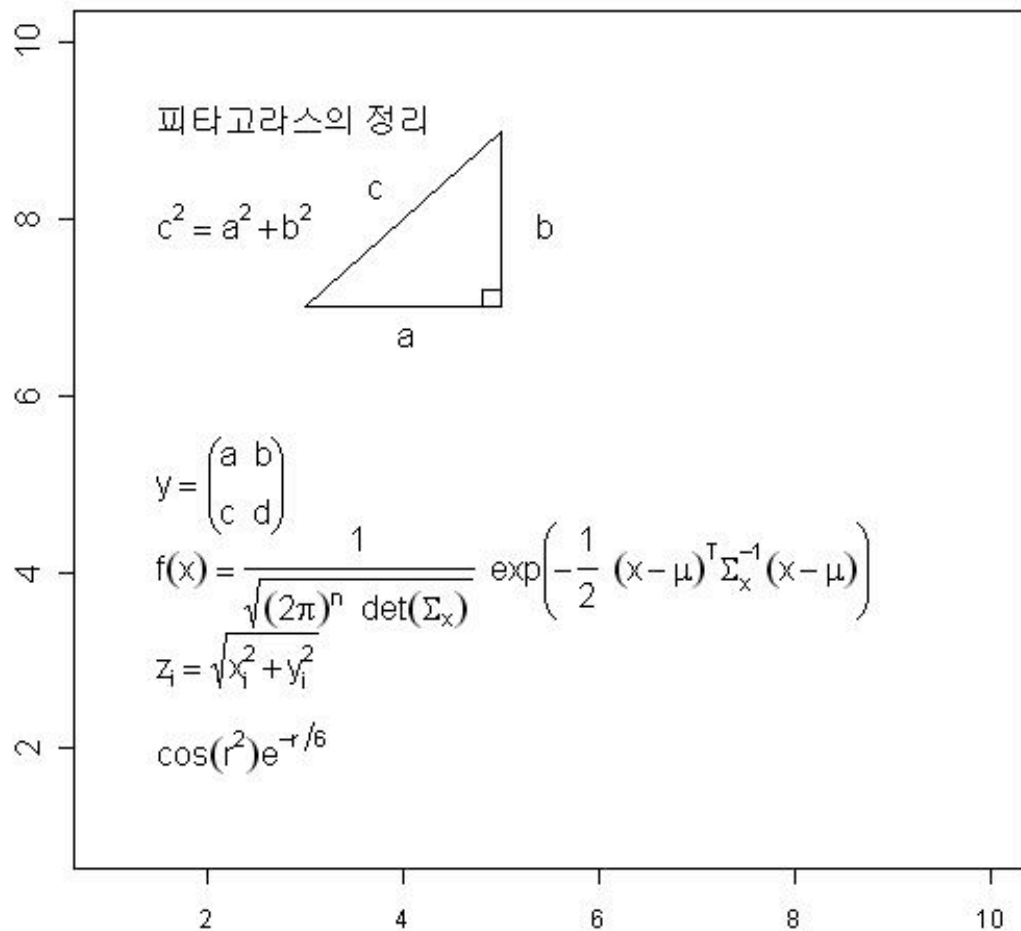

R Font

유 증현

블로그 모음 8탄(<http://blog.naver.com/bdboys>) • (주)오픈베이스 • 2010년 10월 4일



R Font

R Graphics에서 사용할 수 있는 Font를 나열해 보았다. 한글을 위시로한 전각문자를 쓰는 나라에서는 영문을 쓰는 나라보다 불리함 점이 너무 많다. 구현의 문제와 폰트가 이쁘지 않다는 문제가 그것이다. 비단 R뿐만 아니라 컴퓨팅 환경에 두루 해당되는 문제이기도 하다. 한때 R 1.9 버전대에서 한글의 입출력에 문제가 있던 적도 있었다.

```
> jpeg("font.jpg",550,700)
> plot(c(0, 30), c(0, 30), type="n", xlab="", ylab="", axes=FALSE)
> text(10,30, "특수문자 들", col="blue", cex=1.2)
> text(0:25, rep(29,26), paste(0:25), cex=0.7)
> points(0:25, rep(28,26), pch=0:25)
>
> text(10,26, "소문자", col="blue", cex=1.2)
> points(0:25, rep(25,26), pch=letters)
>
> text(10,23, "대문자", col="blue", cex=1.2)
> points(0:25, rep(22,26), pch=LETTERS)
>
> text(10,20, "영문폰트", col="blue", cex=1.2)
> for(i in 1:8){
+   par(font=i)
+   ypos <- 20-i
+   points(0:31, rep(ypos,32), pch=64:95)
+   axis(2, at=ypos, labels=paste("font =",i), las=1)
+ }
>
> letters.kor = c
('가','나','다','라','마','바','사','아','자','차','카','타','파','하')
>
> text(10,10, "한글폰트", col="blue", cex=1.2)
> for(i in 1:10){
+   par(font=i)
+   ypos <- 10-i
```

```

+ points(0:31, rep(ypos,32), pch=letters.kor)
+ axis(2, at=ypos, labels=paste("font =",i), las=1)
+ }
> dev.off()

```

특수문자 들

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
□ ◇ △ + × ◇ ▽ ■ * ♣ ♠ ☼ ☼ ■ ◆ ▲ ◆ ◆ ● ○ ◇ △ ▽

소문자

a b c d e f g h i j k l m n o p q r s t u v w x y z

대문자

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

영문폰트

```

ont = 1 - @ABCDEFGHI JKLMNOPQRSTUVWXYZ [ \ ] ^ -
ont = 2 - @ABCDEFGHI JKLMNOPQRSTUVWXYZ [ \ ] ^ -
ont = 3 - @ABCDEFGHI JKLMNOPQRSTUVWXYZ [ \ ] ^ -
ont = 4 - @ABCDEFGHI JKLMNOPQRSTUVWXYZ [ \ ] ^ -
ont = 5 - ≡ ABXΔEΦΓHI ♯KAMNOΠΘP ΣTY ςΩEΨZ [ : ] ⊥ -
ont = 6 - @ABCDEFGHI JKLMNOPQRSTUVWXYZ [ \ ] ^ -
ont = 7 - @ABCDEFGHI JKLMNOPQRSTUVWXYZ [ \ ] ^ -
ont = 8 - @ABCDEFGHI JKLMNOPQRSTUVWXYZ [ \ ] ^ -

```

한글폰트

```

ont = 1 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라
ont = 2 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라
ont = 3 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라
ont = 4 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라
ont = 5 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라
ont = 6 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라
ont = 7 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라
ont = 8 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라
ont = 9 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라
ont = 10 - 가나다라마바사아자차카타파하가나다라마바사아자차카타파하가나다라

```

다음에는 수학공식의 출력에 대해서 살펴보기로 한다.

수학 공식의 출력

지난번에는 R의 Font에 대해서 간단하게 살펴보았었는데, 이번에는 Graphics Device에 수학 공식을 출력하는 것에 대해서 알아보자.

Chart를 그리고 제목이나 범례, 혹은 각 축의 이름에 일반적인 문자만 사용하라는 법은 없다. R이 통계계산을 위해서 사용되기 위해서는 수학기공식을 출력해야하는 경우도 종종 발생하게 된다. 아니면 좌표 평면에 써 넣어야할 경우도 발생한다. 이 경우에는 text함수를 사용하면된다. 물론 수학기공식을 출력하기 위해서는 각각의 기호에 해당하는 정해진 문법에 따라 기술해 넣으면 된다. 이때 사용하는 함수가 expression이다.

이번 이야기에서는 R의 수학기공식 출력 데모를 보이고, 마지막으로 예제를 보임으로써 같음한다. 자세한 사용법은 expression의 도움말을 참고하기 바란다. 첨언하자면 expression 함수는 수치적분과 미분의 계산할 때도 사용하는 함수이므로, 이 분야에 관심있는 사람은 더 심화학습하기 바란다. 데모에서 왼쪽의 문자를 expression함수의 인자로 사용하면, 오른쪽의 수식으로 출력이된다.

다음의 다섯 그림은 데모이고, 마지막이 이를 응용한 예제이다. 데모를 보기 위해서는 다음과 같이 입력하기만하면 된다.

```
> demo(plotmath)
```

Arithmetic Operators		Radicals	
<code>x + y</code>	$x + y$	<code>sqrt(x)</code>	\sqrt{x}
<code>x - y</code>	$x - y$	<code>sqrt(x, y)</code>	\sqrt{x}
<code>x * y</code>	xy	Relations	
<code>x/y</code>	x/y	<code>x == y</code>	$x = y$
<code>x %+-% y</code>	$x \pm y$	<code>x != y</code>	$x \neq y$
<code>x %/% y</code>	$x + y$	<code>x < y</code>	$x < y$
<code>x %*% y</code>	$x \times y$	<code>x <= y</code>	$x \leq y$
<code>-x</code>	$-x$	<code>x > y</code>	$x > y$
<code>+x</code>	$+x$	<code>x >= y</code>	$x \geq y$
Sub/Superscripts		<code>x %~~% y</code>	$x \approx y$
<code>x[i]</code>	x_i	<code>x %~% y</code>	$x \equiv y$
<code>x^2</code>	x^2	<code>x %==% y</code>	$x \equiv y$
Juxtaposition		<code>x %prop% y</code>	$x \propto y$
<code>x * y</code>	xy	Typeface	
<code>paste(x, y, z)</code>	xyz	<code>plain(x)</code>	x
Lists		<code>italic(x)</code>	x
<code>list(x, y, z)</code>	x, y, z	<code>bold(x)</code>	\mathbf{x}
		<code>bolditalic(x)</code>	\mathbf{x}
		<code>underline(x)</code>	\underline{x}
Ellipsis		Arrows	
<code>list(x[1], ..., x[n])</code>	x_1, \dots, x_n	<code>x %<->% y</code>	$x \leftrightarrow y$
<code>x[1] + ... + x[n]</code>	$x_1 + \dots + x_n$	<code>x %->% y</code>	$x \rightarrow y$
<code>list(x[1], cdots, x[n])</code>	x_1, \dots, x_n	<code>x %<-% y</code>	$x \leftarrow y$
<code>x[1] + ldots + x[n]</code>	$x_1 + \dots + x_n$	<code>x %up% y</code>	$x \uparrow y$
Set Relations		<code>x %down% y</code>	$x \downarrow y$
<code>x %subset% y</code>	$x \subset y$	<code>x %<=>% y</code>	$x \Leftrightarrow y$
<code>x %subsepeq% y</code>	$x \subseteq y$	<code>x %=>% y</code>	$x \Rightarrow y$
<code>x %supset% y</code>	$x \supset y$	<code>x %<=% y</code>	$x \Leftarrow y$
<code>x %supsepeq% y</code>	$x \supseteq y$	<code>x %dblup% y</code>	$x \Uparrow y$
<code>x %notsubset% y</code>	$x \not\subset y$	<code>x %dbldown% y</code>	$x \Downarrow y$
<code>x %in% y</code>	$x \in y$	Symbolic Names	
<code>x %notin% y</code>	$x \notin y$	Alpha - Omega	$A - \Omega$
Accents		alpha - omega	$\alpha - \omega$
<code>hat(x)</code>	\hat{x}	infinity	∞
<code>tilde(x)</code>	\tilde{x}	32 * degree	32°
<code>ring(x)</code>	$\overset{\circ}{x}$	60 * minute	$60'$
<code>bar(xy)</code>	\overline{xy}	30 * second	$30''$
<code>widehat(xy)</code>	\widehat{xy}		
<code>widetilde(xy)</code>	\widetilde{xy}		

Style	
<code>displaystyle(x)</code>	x
<code>textstyle(x)</code>	x
<code>scriptstyle(x)</code>	x
<code>scriptscriptstyle(x)</code>	x
Spacing	
<code>x ~ ~y</code>	$x \ y$

<code>x + phantom(0) + y</code>	$x+ \ +y$
<code>x + over(1, phantom(0))</code>	$x + \frac{1}{-}$

Fractions	
<code>frac(x, y)</code>	$\frac{x}{y}$
<code>over(x, y)</code>	$\frac{x}{y}$
<code>atop(x, y)</code>	$\frac{x}{y}$

Big Operators	
<code>sum(x[i], i = 1, n)</code>	$\sum_{i=1}^n x_i$
<code>prod(plain(P)(X == x), x)</code>	$\prod_x P(X=x)$
<code>integral(f(x) * dx, a, b)</code>	$\int_a^b f(x) dx$
<code>union(A[i], i == 1, n)</code>	$\bigcup_{i=1}^n A_i$
<code>intersect(A[i], i == 1, n)</code>	$\bigcap_{i=1}^n A_i$
<code>lim(f(x), x %->% 0)</code>	$\lim_{x \rightarrow 0} f(x)$
<code>min(g(x), x >= 0)</code>	$\min_{x \geq 0} g(x)$
<code>inf(S)</code>	$\inf S$
<code>sup(S)</code>	$\sup S$

Grouping	
$(x+y)^*z$	$(x+y)z$
x^y+z	x^y+z
$x^{(y+z)}$	$x^{(y+z)}$
$x^{\{y+z\}}$	x^{y+z}
<code>group("(", list(a, b), ")")</code>	$(a, b]$
<code>bggroup("(", atop(x, y), ")")</code>	$\begin{pmatrix} x \\ y \end{pmatrix}$
<code>group(lceil, x, rceil)</code>	$\lceil x \rceil$
<code>group(lfloor, x, rfloor)</code>	$\lfloor x \rfloor$
<code>group(" ", x, " ")</code>	$ x $

마지막으로 위의 내용을 응용한 예제로 오늘의 이야기를 마친다.

```

> plot(1:10,1:10, type="n", xlab="", ylab="")
> text(1.5, 9, adj = 0, labels = "피타고라스의 정리")
> polygon(c(5,3,5),c(9,7,7))
> polygon(c(5,5,4.8,4.8),c(7,7.2,7.2,7))
> text(3.64, 8.36, adj = 0, labels = "c")
> text(3.94, 6.67, adj = 0, labels = "a")
> text(5.36, 7.95, adj = 0, labels = "b")
> text(1.5, 8, adj = 0, labels = expression(c^2 == a^2+b^2))
> text(1.5, 2, adj = 0, labels = expression(cos(r^2) * e^{-r/6}))
> text(1.5, 3, adj = 0, labels = expression(z[i]==sqrt(x[i]^2 + y[i]^2)))
> text(1.5, 4, adj = 0, labels = expression(
+   f(x) == frac(1, sqrt((2 * pi)^n ~ det(Sigma[x]))) ~ exp * bgroup
+   ("(",
+   -frac(1, 2) ~ (x - mu)^T * Sigma[x]^-1 * (x - mu), ")"))
> text(1.5, 5, adj = 0, labels = expression(y==bggroup("(",atop(a ~ b, c
~ d),")")))

```

