Symbol	Usage
~	Separates response variables on the left from the explanatory variables on the right. For example, a prediction of y from x, z, and w would be coded `y ~ x + z + w`.
+	Separates predictor variables.
:	Denotes an interaction between predictor variables. A prediction of y from x, z, and the interaction between x and z would be coded `y ~ x + z + x:z`.
*	A shortcut for denoting all possible interactions. The code `y ~ x * z * w` expands to `y ~ x + z + w + x:z + x:w + z:w + x:z:w`.
^	Denotes interactions up to a specified degree. The code `y ~ (x + z + w)^2` expands to `y ~ x + z + w + x:z + x:w + z:w`.
.	A placeholder for all other variables in the data frame except the dependent variable. For example, if a data frame contained the variables `x, y, z, and w`, then the code `y ~ .` would expand to `y ~ x + z + w`.
-	A minus sign removes a variable from the equation. For example, `y ~ (x + z + w)^2 - x:w` expands to `y ~ x + z + w + x:z + z:w`.
-1	Suppresses the intercept. For example, the formula `y ~ x -1` fits a regression of y on x, and forces the line through the origin at x=0.

Symbol	Usage
`I()`	Elements within the parentheses are interpreted arithmetically. For example, `y ~ x + (z + w)^2` would expand to `y ~ x + z + w + z:w`. In contrast, the code `y ~ x + I((z + w)^2)` would expand to `y ~ x + h`, where `h` is a new variable created by squaring the sum of `z` and `w`.
function	Mathematical functions can be used in formulas. For example, `log(y) ~ x + z + w` would predict `log(y)` from `x`, `z`, and `w`.