

<code>AdjR2(<i>tdf</i>=total.df, null.dev=null.deviance, resid.dev=residual.deviane, k=#indepvars)</code>
<code>CI95(<i>m</i>=mean, <i>se</i>=standard.error); CI99 (<i>m</i>=mean, <i>se</i>=standard.error)</code>
<code>Colors()</code>
<code>compmeans(<i>x</i>=<i>y</i>, <i>f</i>=<i>x</i>, <i>w</i>=<i>w</i>, <i>plot</i>=T/F ...)</code>
<code>CramersV(<i>chi</i>=chi2.statistic, <i>r</i>=#rows, <i>c</i>=#columns, <i>n</i>=sample.size)</code>
<code>crosstab(<i>dep</i>=<i>x</i>, <i>indep</i>=<i>y</i>, <i>weight</i>=<i>w</i>)</code>
<code>csv.get("csv.dataset.csv")</code> [import data in .csv format]
<code>cut2(<i>x</i>=variable, <i>cuts</i>=cutpoints, <i>m</i>=min.obs, <i>g</i>=num.groups ...) [use cuts or g]</code>
<code>ddply(.data, .variables, .function ...)</code> [see help(ddply) for special input format]
<code>describe(<i>x</i>=variable, <i>weights</i>=<i>w</i> ...)</code>
<code>fit.svyglm(<i>svyglm</i>=svyglm.model)</code>

(Continued)