

## 0.1 `plot.ci`: Plotting Vertical confidence Intervals

### Description

The `plot.ci` command generates vertical confidence intervals for linear or generalized linear univariate response models.

### Usage

```
plot.ci(x, CI = 95, qi = "ev", main = "", ylab = NULL, xlab = NULL,  
        xlim = NULL, ylim = NULL, col = c("red", "blue"), ...)
```

### Arguments

<code>x</code>	stored output from <code>sim</code> . The <code>x\$x</code> and optional <code>x\$x1</code> values used to generate the <code>sim</code> output object must have more than one observation.
<code>CI</code>	the selected confidence interval. Defaults to 95 percent.
<code>qi</code>	the selected quantity of interest. Defaults to expected values.
<code>main</code>	a title for the plot.
<code>ylab</code>	label for the y-axis.
<code>xlab</code>	label for the x-axis.
<code>xlim</code>	limits on the x-axis.
<code>ylim</code>	limits on the y-axis.
<code>col</code>	a vector of at most two colors for plotting the expected value given by <code>x</code> and the alternative set of expected values given by <code>x1</code> in <code>sim</code> . If the quantity of interest selected is not the expected value, or <code>x1 = NULL</code> , only the first color will be used.
<code>...</code>	Additional parameters passed to <code>plot</code> .

### Value

For all univariate response models, `plot.ci()` returns vertical confidence intervals over a specified range of one explanatory variable. You may save this plot using the commands described in the Zelig manual (<http://gking.harvard.edu/zelig>).

### Author(s)

Kosuke Imai <[kimai@princeton.edu](mailto:kimai@princeton.edu)>; Gary King <[king@harvard.edu](mailto:king@harvard.edu)>; Olivia Lau <[olau@fas.harvard.edu](mailto:olau@fas.harvard.edu)>

## See Also

The full Zelig manual is available at <http://gking.harvard.edu/zelig>, and users may also wish to see `plot`, `lines`.

## Examples

```
data(turnout)
z.out <- zelig(vote ~ race + educate + age + I(age^2) + income,
              model = "logit", data = turnout)
age.range <- 18:95
x.low <- setx(z.out, educate = 12, age = age.range)
x.high <- setx(z.out, educate = 16, age = age.range)
s.out <- sim(z.out, x = x.low, x1 = x.high)
plot.ci(s.out, xlab = "Age in Years",
        ylab = "Predicted Probability of Voting",
        main = "Effect of Education and Age on Voting Behavior")
legend(45, 0.52, legend = c("College Education (16 years)",
                             "High School Education (12 years)"), col = c("blue", "red"),
       lty = c("solid"))
```