## 0.1 `twosls`: Two Stage Least Squares

`twosls` provides consistent estimates for linear regression models with some explanatory variable correlated with the error term using instrumental variables. In this situation, ordinary least squares fails to provide consistent estimates. The name two-stage least squares stems from the two regressions in the estimation procedure. In stage one, an ordinary least squares prediction of the instrumental variable is obtained from regressing it on the instrument variables. In stage two, the coefficients of interest are estimated using ordinary least square after substituting the instrumental variable by its predictions from stage one.

### Syntax

```
> fml <- list("mu1"  = Y ~ X + W, "mu2" = W ~ X + Z,
              "inst" = ~ X + Z)
> z.out <- zelig(formula = fml, model = "twosls", data = mydata)
> x.out <- setx(z.out)
> s.out <- sim(z.out, x = x.out)
```

### Inputs

`twosls` regression take the following inputs:

- `formula`: A list of the formula for the main equation, the formula for the endogenous variable, and the (one-sided) formula for instrumental variables (including covariates). The first object in the list `mu` corresponds to the main regression model needs to be estimated. Alternatively, a system of simultaneous equations can be used. See the help file of `systemfit` for more information. For example:

  ```
  > fml <- list(mu1 = Y ~ X + W, mu2 = W ~ X + Z, inst = ~X +
  +     Z)
  ```

  - `Y`: the dependent variable of interest.
  - `X`: the covariate.
  - `W`: the endogenous variable.
  - `Z`: the exogenous instrumental variable.

### Additional Inputs

`twosls` takes the following additional inputs for model specifications:

- `TX`: an optional matrix to transform the regressor matrix and, hence, also the coefficient vector (see 0.1). Default is `NULL`.
- `rcovformula`: formula to calculate the estimated residual covariance matrix (see 0.1). Default is equal to 1.

- `probdfsys`: use the degrees of freedom of the whole system (in place of the degrees of freedom of the single equation to calculate probability values for the t-test of individual parameters.
- `single.eq.sigma`: use different $\sigma^2$ for each single equation to calculate the covariance matrix and the standard errors of the coefficients.
- `solvetol`: tolerance level for detecting linear dependencies when inverting a matrix or calculating a determinant. Default is `solvetol`=.Machine$double.eps.
- `saveMemory`: logical. Save memory by omitting some calculation that are not crucial for the basic estimate (e.g McElroy's $R^2$).

**Details**

- `TX`: The matrix `TX` transforms the regressor matrix ($X$) by $X* = X \times TX$. Thus, the vector of coefficients is now $b = TX \times b*$ where $b$ is the original(stacked) vector of all coefficients and $b*$ is the new coefficient vector that is estimated instead. Thus, the elements of vector $b$ and $b_i = \sum_j TX_{ij} \times b_j*$. The $TX$ matrix can be used to change the order of the coefficients and also to restrict coefficients (if $TX$ has less columns than it has rows).
- `rcovformula`: The formula to calculate the estimated covariance matrix of the residuals($\hat{\Sigma}$)can be one of the following (see Judge et al., 1955, p.469): if `rcovformula`= 0:

$$\hat{\sigma_{ij}} = \frac{\hat{e_i}\prime\hat{e_j}}{T}$$

if `rcovformula`= 1 or `rcovformula`='geomean':

$$\hat{\sigma_{ij}} = \frac{\hat{e_i}\prime\hat{e_j}}{\sqrt{(T-k_i)\times(T-k_j)}}$$

if `rcovformula`= 2 or `rcovformula`='Theil':

$$\hat{\sigma_{ij}} = \frac{\hat{e_i}\prime\hat{e_j}}{T-k_i-k_j+tr[X_i(X_i\prime X_i)^{-1}X_i\prime X_j(X_j\prime X_j)^{-1}X_j\prime]}$$

if `rcovformula`= 3 or `rcovformula`='max':

$$\hat{\sigma_{ij}} = \frac{\hat{e_i}\prime\hat{e_j}}{T-max(k_i,k_j)}$$

If $i = j$, formula 1, 2, and 3 are equal. All these three formulas yield unbiased estimators for the diagonal elements of the residual covariance matrix. If $ineqj$, only formula 2 yields an unbiased estimator for the residual covariance matrix, but it is not necessarily positive semidefinit. Thus, it is doubtful whether formula 2 is really superior to formula 1

**Examples**

Attaching the example dataset:

```
> data(klein)
```

Formula:

```
> formula <- list(mu1 = C ~ Wtot + P + P1, mu2 = I ~ P + P1 +
+     K1, mu3 = Wp ~ X + X1 + Tm, inst = ~P1 + K1 + X1 + Tm +
+     Wg + G)
```

Estimating the model using `twosls`:

```
> z.out <- zelig(formula = formula, model = "twosls", data = klein)
> summary(z.out)
```

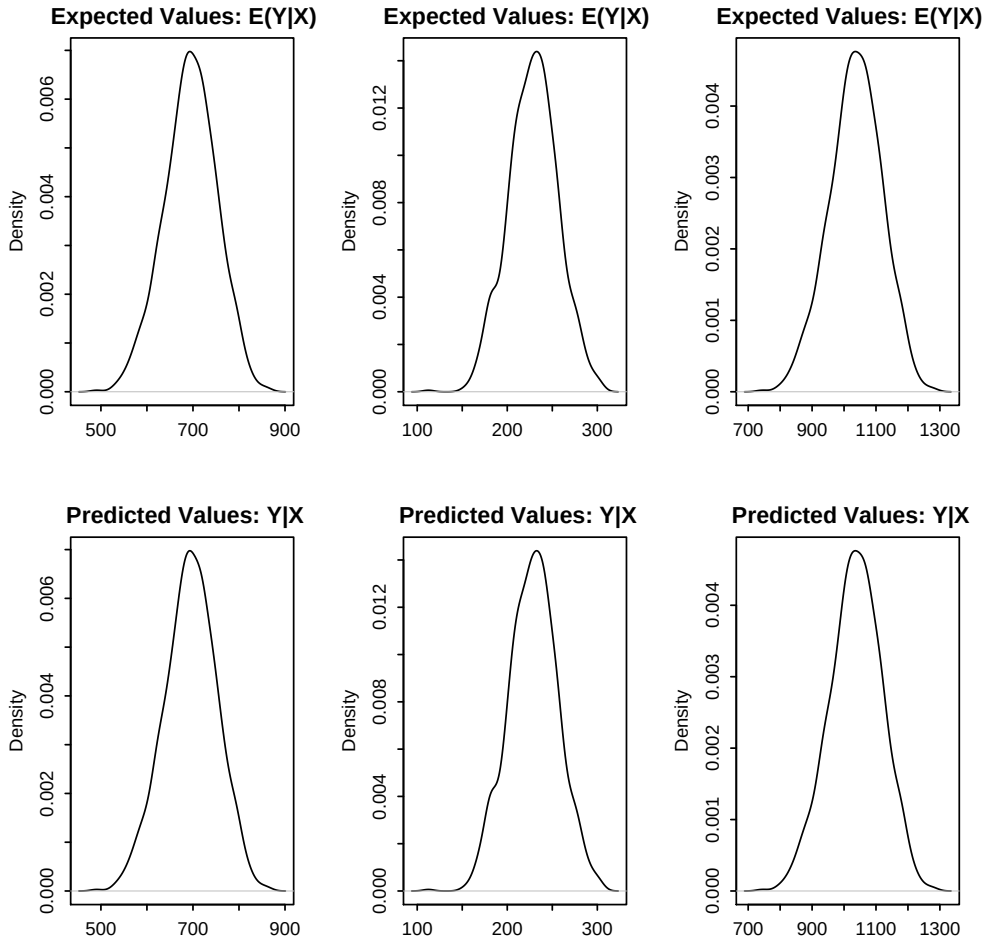
Set explanatory variables to their default (mean/mode) values

```
> x.out <- setx(z.out)
```

Simulate draws from the posterior distribution:

```
> s.out <- sim(z.out, x = x.out)
> summary(s.out)
```

Plot the quantities of interest


Expected Values: E(Y|X)
Density
0.000
0.002
0.004
0.006
500
700
900
Expected Values: E(Y|X)
Density
0.000
0.004
0.008
0.012
100
200
300
Expected Values: E(Y|X)
Density
0.000
0.001
0.002
0.003
0.004
700
900
1100
1300
Predicted Values: Y|X
Density
0.000
0.002
0.004
0.006
500
700
900
Predicted Values: Y|X
Density
0.000
0.004
0.008
0.012
100
200
300
Predicted Values: Y|X
Density
0.000
0.001
0.002
0.003
0.004
700
900
1100
1300

### Model

Let's consider the following regression model,

$$Y_i = X_i\beta + Z_i\gamma + \epsilon_i, \quad i = 1, \ldots, N$$

where $Y_i$ is the dependent variable, $X_i = (X_{1i}, \ldots, X_{Ni})$ is the vector of explanatory variables, $\beta$ is the vector of coefficients of the explanatory variables $X_i$, $Z_i$ is the problematic explanatory variable, and $\gamma$ is the coefficient of $Z_i$. In the equation, there is a direct dependence of $Z_i$ on the structural disturbances of $\epsilon$.

- The *stochastic component* is given by

$$\epsilon_i \sim \mathcal{N}(0, \sigma^2), \quad \text{and} \quad \text{cov}(Z_i, \epsilon_i) \neq 0,$$

- The *systematic component* is given by:

$$\mu_i = E(Y_i) = X_i\beta + Z_i\gamma,$$

To correct the problem caused by the correlation of $Z_i$ and $\epsilon$, two stage least squares utilizes two steps:

- *Stage 1*: A new instrumental variable $\hat{Z}$ is created for $Z_i$ which is the ordinary least squares predictions from regressing $Z_i$ on a set of exogenous instruments $W$ and $X$.

$$\widehat{Z}_i = \widetilde{W}_i[(\widetilde{W}^\top\widetilde{W})^{-1}\widetilde{W}^\top Z]$$

  where $\widetilde{W} = (W, X)$

- *Stage 2*: Substitute for $\hat{Z}_i$ for $Z_i$ in the original equation, estimate $\beta$ and $\gamma$ by ordinary least squares regression of $Y$ on $X$ and $\hat{Z}$ as in the following equation.

$$Y_i = X_i\beta + \widehat{Z}_i\gamma + \epsilon_i, \quad \text{for} \quad i = 1, \ldots, N$$

### See Also

For information about three stage least square regression, see Section **??** and `help(3sls)`. For information about seemingly unrelated regression, see Section **??** and `help(sur)`.

### Quantities of Interest

### Output Values

The output of each Zelig command contains useful information which you may view. For example, if you run:

```
z.out <- zelig(formula=fml, model = "twosls", data)
```

then you may examine the available information in `z.out` by using `names(z.out)`, see the draws from the posterior distribution of the `coefficients` by using `z.out$coefficients`, and view a default summary of information through `summary(z.out)`. Other elements available through the `$` operator are listed below:

- `h`: matrix of all (diagonally stacked) instrumental variables.
- `single.eq.sigma`: different $\sigma^2$s for each single equation?.
- `zelig.data`: the input data frame if `save.data = TRUE`.
- `method`: Estimation method.
- `g`: number of equations.
- `n`: total number of observations.
- `k`: total number of coefficients.
- `ki`: total number of linear independent coefficients.
- `df`: degrees of freedom of the whole system.
- `iter`: number of iteration steps.
- `b`: vector of all estimated coefficients.
- `t`: $t$ values for $b$.
- `se`: estimated standard errors of $b$.
- `bt`: coefficient vector transformed by $TX$.
- `p`: $p$ values for $b$.
- `bcov`: estimated covariance matrix of $b$.
- `btcov`: covariance matrix of $bt$.
- `rcov`: estimated residual covariance matrix.
- `drcov`: determinant of `rcov`.
- `rcor`: estimated residual correlation matrix.
- `olsr2`: system OLS R-squared value.
- `y`: vector of all (stacked) endogenous variables.
- `x`: matrix of all (diagonally stacked) regressors.

- `data`: data frame of the whole system (including instruments).
- `TX`: matrix used to transform the regressor matrix.
- `rcovformula`: formula to calculate the estimated residual covariance matrix.
- `probdfsys`: system degrees of freedom to calculate probability values?.
- `solvetol`: tolerance level when inverting a matrix or calculating a determinant.
- `eq`: a list that contains the results that belong to the individual equations.
- `eqnlabel*`: the equation label of the ith equation (from the labels list).
- `formula*`: model formula of the ith equation.
- `n*`: number of observations of the ith equation.
- `k*`: number of coefficients/regressors in the ith equation (including the constant).
- `ki*`: number of linear independent coefficients in the ith equation (including the constant differs from k only if there are restrictions that are not cross equation).
- `df*`: degrees of freedom of the ith equation.
- `b*`: estimated coefficients of the ith equation.
- `se*`: estimated standard errors of $b$ of the ith equation.
- `t*`: $t$ values for $b$ of the ith equation.
- `p*`: $p$ values for $b$ of the ith equation.
- `covb*`: estimated covariance matrix of $b$ of the ith equation.
- `y*`: vector of endogenous variable (response values) of the ith equation.
- `x*`: matrix of regressors (model matrix) of the ith equation.
- `data*`: data frame (including instruments) of the ith equation.
- `fitted*`: vector of fitted values of the ith equation.
- `residuals*`: vector of residuals of the ith equaiton.
- `ssr*`: sum of squared residuals of the ith equation.
- `mse*`: estimated variance of the residuals (mean of squared errors) of the ith equation.
- `s2*`: estimated variance of the residents($\hat{sigma}^2$) of the ith equation.

- `rmse*`: estimated standard error of the reiduals (square root of mse) of the ith equation.
- `s*`: estimated standard error of the residuals ($\hat{\sigma}$) of the ith equation.
- `r2*`: R-squared (coefficient of determination).
- `adjr2*`: adjusted R-squared value.
- `inst*`: instruments of the ith equation.
- `h*`: matrix of instrumental variables of the ith equation.

## How to Cite

To cite the *twosls* Zelig model:

> Ferdinand Alimadhi, Ying Lu, and Elena Villalon. 2007. "twosls: Two Stage Least Squares," in Kosuke Imai, Gary King, and Olivia Lau, "Zelig: Everyone's Statistical Software," `http://gking.harvard.edu/zelig`.

To cite Zelig as a whole, please reference these two sources:

> Kosuke Imai, Gary King, and Olivia Lau. 2007. "Zelig: Everyone's Statistical Software," `http://GKing.harvard.edu/zelig`.

> Kosuke Imai, Gary King, and Olivia Lau. 2008. "Toward A Common Framework for Statistical Analysis and Development," *Journal of Computational and Graphical Statistics*, forthcoming, `http://gking.harvard.edu/files/abs/z-abs.shtml`.

## See also

The twosls function is adapted from the `systemfit` library (Hamann and Henningsen 2005).

# Bibliography

Hamann, J. and Henningsen, A. (2005), *systemfit: Simultaneous Equation Systems in R Package*.