statsmodels.regression.linear_model.RegressionResults

class `statsmodels.regression.linear_model.RegressionResults`(*model*, *params*, *normalized_cov_params=None*, *scale=1.0*, *cov_type='nonrobust'*, *cov_kwds=None*, *use_t=None*, ***kwargs*)[source] [../_modules/statsmodels/regression/linear_model.html#RegressionResults]

This class summarizes the fit of a linear regression model.

It handles the output of contrasts, estimates of covariance, etc.

Parameters

model : `RegressionModel`

The regression model instance.

params : `ndarray` [https://docs.scipy.org/doc/numpy/reference/generated/numpy.ndarray.html#numpy.ndarray]

The estimated parameters.

normalized_cov_params : `ndarray` [https://docs.scipy.org/doc/numpy/reference/generated/numpy.ndarray.html#numpy.ndarray]

The normalized covariance parameters.

scale : `float` [https://docs.python.org/3/library/functions.html#float]

The estimated scale of the residuals.

cov_type : `str` [https://docs.python.org/3/library/stdtypes.html#str]

The covariance estimator used in the results.

cov_kwds : `dict` [https://docs.python.org/3/library/stdtypes.html#dict]

Additional keywords used in the covariance specification.

use_t : bool [https://docs.python.org/3/library/stdtypes.html#bltin-boolean-values]

Flag indicating to use the Student’s t in inference.

****kwargs**

Additional keyword arguments used to initialize the results.

Attributes

pinv_wexog

See model class docstring for implementation details.

cov_type

Parameter covariance estimator used for standard errors and t-stats.

df_model

Model degrees of freedom. The number of regressors *p*. Does not include the constant if one is present.

df_resid

Residual degrees of freedom. *n - p - 1*, if a constant is present. *n - p* if a constant is not included.

het_scale

adjusted squared residuals for heteroscedasticity robust standard errors. Is only available after *HC#_se* or *cov_HC#* is called. See HC#_se for more information.

history

Estimation history for iterative estimators.

model

A pointer to the model instance that called fit() or results.

params

The linear coefficients that minimize the least squares criterion. This is usually called Beta for the classical linear model.

Methods

`compare_f_test` [generated/statsmodels.regression.linear_model.RegressionResults.compare_f_test.html#statsmodels.regression.linear_model.RegressionResults.compare_f_test](restricted)	Use F test to te whether restric model is correc
`compare_lm_test` [generated/statsmodels.regression.linear_model.RegressionResults.compare_lm_test.html#statsmodels.regression.linear_model.RegressionResults.compare_lm_test](restricted[, demean, use_	Use Lagrange Multiplier test t test a set of lin restrictions.
`compare_lr_test` [generated/statsmodels.regression.linear_model.RegressionResults.compare_lr_test.html#statsmodels.regression.linear_model.RegressionResults.compare_lr_test](restricted[, large_sample])	Likelihood ratio to test whether restricted mode correct.
`conf_int` [generated/statsmodels.regression.linear_model.RegressionResults.conf_int.html#statsmodels.regression.linear_model.RegressionResults.conf_int]([alpha, cols])	Compute the confidence inte of the fitted parameters.
`cov_params` [generated/statsmodels.regression.linear_model.RegressionResults.cov_params.html#statsmodels.regression.linear_model.RegressionResults.cov_params]([r_matrix, column, scale, cov_p, …])	Compute the variance/covar matrix.

`f_test` [generated/statsmodels.regression.linear_model.RegressionResults.f_test.html#statsmodels.regression.linear_model.RegressionResults.f_test](r_matrix[, cov_p, scale, invcov])	Compute the F-for a joint linea hypothesis.
`get_prediction` [generated/statsmodels.regression.linear_model.RegressionResults.get_prediction.html#statsmodels.regression.linear_model.RegressionResults.get_prediction]([exog, transform, weights, ...])	Compute predi results.
`get_robustcov_results` [generated/statsmodels.regression.linear_model.RegressionResults.get_robustcov_results.html#statsmodels.regression.linear_model.RegressionResults.get_robustcov_r	Create new res instance with robust covariar as default.
`initialize` [generated/statsmodels.regression.linear_model.RegressionResults.initialize.html#statsmodels.regression.linear_model.RegressionResults.initialize](model, params, **kwargs)	Initialize (possi re-initialize) a Results instanc
`load` [generated/statsmodels.regression.linear_model.RegressionResults.load.html#statsmodels.regression.linear_model.RegressionResults.load](fname)	Load a pickled results instanc
`normalized_cov_params` [generated/statsmodels.regression.linear_model.RegressionResults.normalized_cov_params.html#statsmodels.regression.linear_model.RegressionResults.normalized_cov_params]()	See specific mo class docstring
`predict` [generated/statsmodels.regression.linear_model.RegressionResults.predict.html#statsmodels.regression.linear_model.RegressionResults.predict]([exog, transform])	Call self.model.prec with self.param the first argum
`remove_data` [generated/statsmodels.regression.linear_model.RegressionResults.remove_data.html#statsmodels.regression.linear_model.RegressionResults.remove_data]()	Remove data a all nobs arrays result and mod
`save` [generated/statsmodels.regression.linear_model.RegressionResults.save.html#statsmodels.regression.linear_model.RegressionResults.save](fname[, remove_data])	Save a pickle o instance.
`scale` [generated/statsmodels.regression.linear_model.RegressionResults.scale.html#statsmodels.regression.linear_model.RegressionResults.scale]()	A scale factor f the covariance matrix.
`summary` [generated/statsmodels.regression.linear_model.RegressionResults.summary.html#statsmodels.regression.linear_model.RegressionResults.summary]([yname, xname, title, alpha])	Summarize the Regression Res
`summary2` [generated/statsmodels.regression.linear_model.RegressionResults.summary2.html#statsmodels.regression.linear_model.RegressionResults.summary2]([yname, xname, title, alpha, ...])	Experimental summary funct to summarize t regression resu
`t_test` [generated/statsmodels.regression.linear_model.RegressionResults.t_test.html#statsmodels.regression.linear_model.RegressionResults.t_test](r_matrix[, cov_p, scale, use_t])	Compute a t-te a each linear hypothesis of t form Rb = q.
`t_test_pairwise` [generated/statsmodels.regression.linear_model.RegressionResults.t_test_pairwise.html#statsmodels.regression.linear_model.RegressionResults.t_test_pairwise](term_name[, method, alpha, ...])	Perform pairwi t_test with mult testing correcte p-values.

`wald_test` [generated/statsmodels.regression.linear_model.RegressionResults.wald_test.html#statsmodels.regression.linear_model.RegressionResults.wald_test](r_matrix[, cov_p, scale, invcov, …])	Compute a Wal test for a joint l hypothesis.
`wald_test_terms` [generated/statsmodels.regression.linear_model.RegressionResults.wald_test_terms.html#statsmodels.regression.linear_model.RegressionResults.wald_test_terms]([skip_single, …])	Compute a sequence of W tests for terms multiple colum

Properties

`HC0_se` [generated/statsmodels.regression.linear_model.RegressionResults.HC0_se.html#statsmodels.regression.linear_model.RegressionResults.HC0_se]	White’s (1980) heteroskedasticity robust standard errors.
`HC1_se` [generated/statsmodels.regression.linear_model.RegressionResults.HC1_se.html#statsmodels.regression.linear_model.RegressionResults.HC1_se]	MacKinnon and White’s (1985) heteroskedasticity robust standard errors.
`HC2_se` [generated/statsmodels.regression.linear_model.RegressionResults.HC2_se.html#statsmodels.regression.linear_model.RegressionResults.HC2_se]	MacKinnon and White’s (1985) heteroskedasticity robust standard errors.
`HC3_se` [generated/statsmodels.regression.linear_model.RegressionResults.HC3_se.html#statsmodels.regression.linear_model.RegressionResults.HC3_se]	MacKinnon and White’s (1985) heteroskedasticity robust standard errors.
`aic` [generated/statsmodels.regression.linear_model.RegressionResults.aic.html#statsmodels.regression.linear_model.RegressionResults.aic]	Akaike’s information criteria.
`bic` [generated/statsmodels.regression.linear_model.RegressionResults.bic.html#statsmodels.regression.linear_model.RegressionResults.bic]	Bayes’ information criteria.
`bse` [generated/statsmodels.regression.linear_model.RegressionResults.bse.html#statsmodels.regression.linear_model.RegressionResults.bse]	The standard errors of the parameter estimates.
`centered_tss` [generated/statsmodels.regression.linear_model.RegressionResults.centered_tss.html#statsmodels.regression.linear_model.RegressionResults.centered_tss]	The total (weighted) sum of squares centered about the mean.
`condition_number` [generated/statsmodels.regression.linear_model.RegressionResults.condition_number.html#statsmodels.regression.linear_model.RegressionResults.condition_number]	Return condition number of exogenous matrix.
`cov_HC0` [generated/statsmodels.regression.linear_model.RegressionResults.cov_HC0.html#statsmodels.regression.linear_model.RegressionResults.cov_HC0]	Heteroscedasticity robust covariance matrix.
`cov_HC1` [generated/statsmodels.regression.linear_model.RegressionResults.cov_HC1.html#statsmodels.regression.linear_model.RegressionResults.cov_HC1]	Heteroscedasticity robust covariance matrix.
`cov_HC2` [generated/statsmodels.regression.linear_model.RegressionResults.cov_HC2.html#statsmodels.regression.linear_model.RegressionResults.cov_HC2]	Heteroscedasticity robust covariance matrix.
`cov_HC3` [generated/statsmodels.regression.linear_model.RegressionResults.cov_HC3.html#statsmodels.regression.linear_model.RegressionResults.cov_HC3]	Heteroscedasticity robust covariance matrix.
`eigenvals` [generated/statsmodels.regression.linear_model.RegressionResults.eigenvals.html#statsmodels.regression.linear_model.RegressionResults.eigenvals]	Return eigenvalues sorted in decreasing order.

`ess` [generated/statsmodels.regression.linear_model.RegressionResults.ess.html#statsmodels.regression.linear_model.RegressionResults.ess]	The explained sum of squares.
`f_pvalue` [generated/statsmodels.regression.linear_model.RegressionResults.f_pvalue.html#statsmodels.regression.linear_model.RegressionResults.f_pvalue]	The p-value of the F-statistic.
`fittedvalues` [generated/statsmodels.regression.linear_model.RegressionResults.fittedvalues.html#statsmodels.regression.linear_model.RegressionResults.fittedvalues]	The predicted values for the original (unwhitened) design.
`fvalue` [generated/statsmodels.regression.linear_model.RegressionResults.fvalue.html#statsmodels.regression.linear_model.RegressionResults.fvalue]	F-statistic of the fully specified model.
`llf` [generated/statsmodels.regression.linear_model.RegressionResults.llf.html#statsmodels.regression.linear_model.RegressionResults.llf]	Log-likelihood of model
`mse_model` [generated/statsmodels.regression.linear_model.RegressionResults.mse_model.html#statsmodels.regression.linear_model.RegressionResults.mse_model]	Mean squared error the model.
`mse_resid` [generated/statsmodels.regression.linear_model.RegressionResults.mse_resid.html#statsmodels.regression.linear_model.RegressionResults.mse_resid]	Mean squared error of the residuals.
`mse_total` [generated/statsmodels.regression.linear_model.RegressionResults.mse_total.html#statsmodels.regression.linear_model.RegressionResults.mse_total]	Total mean squared error.
`nobs` [generated/statsmodels.regression.linear_model.RegressionResults.nobs.html#statsmodels.regression.linear_model.RegressionResults.nobs]	Number of observations n.
`pvalues` [generated/statsmodels.regression.linear_model.RegressionResults.pvalues.html#statsmodels.regression.linear_model.RegressionResults.pvalues]	The two-tailed p values for the t-stats of the params.
`resid` [generated/statsmodels.regression.linear_model.RegressionResults.resid.html#statsmodels.regression.linear_model.RegressionResults.resid]	The residuals of the model.
`resid_pearson` [generated/statsmodels.regression.linear_model.RegressionResults.resid_pearson.html#statsmodels.regression.linear_model.RegressionResults.resid_pearson]	Residuals, normalized to have unit variance.
`rsquared` [generated/statsmodels.regression.linear_model.RegressionResults.rsquared.html#statsmodels.regression.linear_model.RegressionResults.rsquared]	R-squared of the model.
`rsquared_adj` [generated/statsmodels.regression.linear_model.RegressionResults.rsquared_adj.html#statsmodels.regression.linear_model.RegressionResults.rsquared_adj]	Adjusted R-squared.
`ssr` [generated/statsmodels.regression.linear_model.RegressionResults.ssr.html#statsmodels.regression.linear_model.RegressionResults.ssr]	Sum of squared (whitened) residuals.
`tvalues` [generated/statsmodels.regression.linear_model.RegressionResults.tvalues.html#statsmodels.regression.linear_model.RegressionResults.tvalues]	Return the t-statistic for a given parameter estimate.
`uncentered_tss` [generated/statsmodels.regression.linear_model.RegressionResults.uncentered_tss.html#statsmodels.regression.linear_model.RegressionResults.uncentered_tss]	Uncentered sum of squares.
`use_t` [generated/statsmodels.regression.linear_model.RegressionResults.use_t.html#statsmodels.regression.linear_model.RegressionResults.use_t]	Flag indicating to use the Student’s distribution in inference.
`wresid` [generated/statsmodels.regression.linear_model.RegressionResults.wresid.html#statsmodels.regression.linear_model.RegressionResults.wresid]	The residuals of the transformed/whitened regressand and regressor(s).