Python's statsmodels package: OLS vs. WLS

Yan Zeng

Version 1.0, last updated on 2021-04-12

Abstract

Clarify the difference between statsmodels.WLS and statsmodels.OLS.

In the documentation of the class statsmodels.regression.linear_models.WLS [1], it is stated that

class `statsmodels.regression.linear_model.WLS`(*endog, exog, weights=1.0, missing='none', hasconst=None, **kwargs*)[source]

Weighted Least Squares

The weights are presumed to be (proportional to) the inverse of the variance of the observations. That is, if the variables are to be transformed by 1/sqrt(W) you must supply weights = 1/W.

The phrase "if the variables are to be transformed by `1/sqrt(W)` you must supply `weights = 1/W`" should be corrected to "if the variables are to be transformed by `sqrt(W)` you must supply `weights = W`".

The reason is that when we assume $W \sim \frac{1}{\sigma_Y^2}$, we will have

$$\sigma^2(\sqrt{W}Y) = W \cdot \sigma_Y^2 \sim \text{const}$$

So, it's the weights that are to be multiplied to the endogenous and exogenous variables, not the inverse of the weights.

To verify this numerically, for regression $Y = \beta X + \varepsilon$, we can compare the results of running the following two commands:

- `statsmodels.WLS(endog=`Y`, exog=`X`, weights=`w`)`
- `statsmodels.OLS(endog=`$\sqrt{w}Y$`, exog=`$\sqrt{w}X$`, weights=1)`

We can verify that

$$\beta^{WLS} = \beta^{OLS},\ \varepsilon^{WLS} \cdot \sqrt{w} = \varepsilon^{OLS}$$

So, we can conclude what the code is doing under the hood:

$$\texttt{WLS}: \sqrt{w}Y = \sqrt{w}X\beta^{WLS} + \sqrt{w}\varepsilon^{WLS},\ \texttt{OLS}: \sqrt{w}Y = \sqrt{w}X\beta^{OLS} + \varepsilon^{OLS}$$

References

[1] statsmodels v0.12.2: `Documentation of statsmodels.regression.linear_model.WLS`.