

How do you perform White's test in Python step-by-step?

Authored by
stats writer

July 1, 2024

RECOMMENDED CITATION

stats writer (2024). *How do you perform White's test in Python step-by-step?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165400>

White's test is a statistical method used to evaluate the homoscedasticity (equal variance) assumption in a linear regression model. This test helps determine if the variance of the residuals is constant across all values of the independent variables. In Python, performing White's test involves the following steps:

1. Import necessary packages: The first step is to import the necessary packages such as statsmodels and numpy.
2. Load the data: Load the data set into your Python environment that contains the dependent variable and the independent variables.
3. Fit the OLS regression model: Use the statsmodels package to fit an ordinary least squares (OLS) regression model with the dependent variable and independent variables.
4. Obtain the residuals: Once the model has been fitted, obtain the residuals (i.e. the difference between the actual and predicted values) from the model.
5. Create a new dataframe: Create a new dataframe with the squared values of the independent variables and the cross-product of the independent variables.
6. Fit a new OLS model: Use the statsmodels package to fit a new OLS model with the squared values of independent variables and the cross-product of the independent variables as the independent variables, and the residuals as the dependent variable.
7. Perform the test: The next step is to perform the White's test using the results from the second OLS model. The test statistic is calculated by taking the sum of squared residuals from the second model and dividing it by the sum of squared residuals from the first model. This statistic follows a chi-square distribution with degrees of freedom equal to the number of independent variables in the second model.
8. Interpret the results: The final step is to interpret the results of the White's test. If the p-value associated with the test statistic is greater than the chosen significance level (usually 0.05), then we fail to reject the null hypothesis, which suggests that the variance is constant (homoscedasticity is present) in the linear regression model.

In summary, White's test in Python involves fitting two OLS models and performing a chi-square test on the residuals to determine if the homoscedasticity assumption is satisfied in the linear regression model.

Perform White's Test in Python (Step-by-Step)

White's test is used to determine if is present in a regression model.

Heteroscedasticity refers to the unequal scatter of at different levels of a , which violates the that the residuals are equally scattered at each level of the response variable.

The following step-by-step example shows how to perform White's test in Python to determine whether or not heteroscedasticity is a problem in a given regression model.

Step 1: Load Data

In this example we will fit a using the mtcars dataset.

The following code shows how to load this dataset into a pandas DataFrame:

```
from sklearn.linear_model import LinearRegression
from statsmodels.stats.diagnostic import het_white
import statsmodels.api as sm
import pandas as pd
```

#define URL where dataset is located

```
url =  
"https://raw.githubusercontent.com/Statology/Python-G  
uides/main/mtcars.csv"
```

#read in data

```
data = pd.read_csv(url)
```

#view summary of data

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 32 entries, 0 to 31
```

```
Data columns (total 12 columns):
```

```
# Column Non-Null Count Dtype
```

```
--- -----
```

```
0 model 32 non-null object
```

```
1 mpg 32 non-null float64
```

```
2 cyl 32 non-null int64
```

```
3 disp 32 non-null float64
```

```
4 hp 32 non-null int64
```

```
5 drat 32 non-null float64
```

```
6 wt 32 non-null float64
```

```
7 qsec 32 non-null float64
```

```
8 vs 32 non-null int64
```

```
9 am 32 non-null int64
10 gear 32 non-null int64
11 carb 32 non-null int64
dtypes: float64(5), int64(6), object(1)
```

Step 2: Fit Regression Model

Next, we will fit a regression model using mpg as the response variable and disp and hp as the two predictor variables:

```
#define response variable
y = data

#define predictor variables
x = data]

#add constant to predictor variables
x = sm.add_constant(x)

#fit regression model
model = sm.OLS(y, x).fit()
```

Step 3: Perform White's Test

Next, we will use the function from the statsmodels

package to perform White's test to determine if heteroscedasticity is present in the regression model:

```
#perform White's test
```

```
white_test = het_white(model.resid, model.model.exog)
```

```
#define labels to use for output of White's test
```

```
labels =
```

```
#print results of White's test
```

```
print(dict(zip(labels, white_test)))
```

```
{'Test Statistic': 7.076620330416624, 'Test Statistic p-value': 0.21500404394263936, 'F-Statistic': 1.4764621093131864, 'F-Test p-value': 0.23147065943879694}
```

Here is how to interpret the output:

The test statistic is $X^2 = 7.0766$. The corresponding p-value is 0.215.

White's test uses the following null and alternative hypotheses:

Null (H_0): Homoscedasticity is present (residuals are

equally scattered) Alternative (HA): Heteroscedasticity is present (residuals are not equally scattered)

This means we do not have sufficient evidence to say that heteroscedasticity is present in the regression model.

What To Do Next

If you fail to reject the null hypothesis of White's test then heteroscedasticity is not present and you can proceed to interpret the output of the original regression.

However, if you reject the null hypothesis, this means heteroscedasticity is present. In this case, the standard errors that are shown in the output table of the regression may be unreliable.

There are two common ways to fix this issue:

1. Transform the response variable.

You can try performing a transformation on the response variable, such as taking of the response variable. This often causes heteroscedasticity to go

away.

2. Use weighted regression.

Weighted regression assigns a weight to each data point based on the variance of its fitted value. Essentially, this gives small weights to data points that have higher variances, which shrinks their squared residuals. When the proper weights are used, this can eliminate the problem of heteroscedasticity.

Additional Resources

The following tutorials provide additional information about linear regression in Python: