NOVA IMS Information Management School

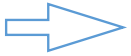
Programming for Data Science
Lecture 6

Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration

*because R is for nerds

Statsmodels

Load statsmodel 

```
import statsmodels.api as sm
```

Statsmodels

Load statsmodel

```
import statsmodels.api as sm
```

Load sample dataset

```
df = sm.datasets.get_rdataset("Guerry", "HistData").data
```

Statsmodels

Load statsmodel

```
import statsmodels.api as sm
```

Load sample dataset

```
df = sm.datasets.get_rdataset("Guerry", "HistData").data
```

It is a Pandas Data Frame hurray!

```
df.head()
```

	dept	Region	Department	Crime_pers	Crime_prop	Literacy	Donations	Infants	Suicides	MainCity	...	Crime_parents	Infanticide	Donation_clergy	Lottery
0	1	E	Ain	28870	15890	37	5098	33120	35039	2:Med	...	71	60	69	41
1	2	N	Aisne	26226	5521	51	8901	14572	12831	2:Med	...	4	82	36	38
2	3	C	Allier	26747	7925	13	10973	17044	114121	2:Med	...	46	42	76	66
3	4	E	Basses-Alpes	12935	7289	46	2733	23018	14238	1:Sm	...	70	12	37	80
4	5	E	Hautes-Alpes	17488	8174	69	6962	23076	16171	1:Sm	...	22	23	64	79

5 rows × 23 columns

Statsmodels

Load statsmodel

```
import statsmodels.api as sm
```

Load sample dataset

```
df = sm.datasets.get_rdataset("Guerry", "HistData").data
```

Filter Data, Select a few features

```
vars = ['Department', 'Lottery', 'Literacy', 'Wealth', 'Region']
```

```
df = df[vars]
```

```
df.head()
```

	Department	Lottery	Literacy	Wealth	Region
0	Ain	41	37	73	E
1	Aisne	38	51	22	N
2	Allier	66	13	61	C
3	Basses-Alpes	80	46	76	E
4	Hautes-Alpes	79	69	83	E

Statsmodels

Filter Data, Select a few features

```
df.head()
```

	Department	Lottery	Literacy	Wealth	Region
0	Ain	41	37	73	E
1	Aisne	38	51	22	N
2	Allier	66	13	61	C
3	Basses-Alpes	80	46	76	E
4	Hautes-Alpes	79	69	83	E

Drop NaN entries

```
df = df.dropna()
```

Build Design Matrix we use patsy for this

```
from patsy import dmatrices
y, X = dmatrices(
    'Lottery ~ Literacy + Wealth + Region',
    data=df,
    return_type='dataframe'
)
```

Statsmodels

Design Matrix

Consider the following Linear Model $$y_i = \beta_0 + \beta_1 w_i + \beta_2 x_i + \varepsilon_i$$

is equivalent to

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 1 & w_1 & x_1 \\ 1 & w_2 & x_2 \\ 1 & w_3 & x_3 \\ 1 & w_4 & x_4 \\ 1 & w_5 & x_5 \\ 1 & w_6 & x_6 \\ 1 & w_7 & x_7 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \\ \varepsilon_7 \end{bmatrix}$$

Statsmodels

Design Matrix

Consider the following Linear Model $$y_i = \beta_0 + \beta_1 w_i + \beta_2 x_i + \varepsilon_i$$

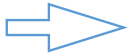
is equivalent to

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 1 & w_1 & x_1 \\ 1 & w_2 & x_2 \\ 1 & w_3 & x_3 \\ 1 & w_4 & x_4 \\ 1 & w_5 & x_5 \\ 1 & w_6 & x_6 \\ 1 & w_7 & x_7 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \\ \varepsilon_7 \end{bmatrix}$$

Design Matrix

Statsmodels

Resulting Design Matrix



```
X.head()
```

	Intercept	Region[T.E]	Region[T.N]	Region[T.S]	Region[T.W]	Literacy	Wealth
0	1.0	1.0	0.0	0.0	0.0	37.0	73.0
1	1.0	0.0	1.0	0.0	0.0	51.0	22.0
2	1.0	0.0	0.0	0.0	0.0	13.0	61.0
3	1.0	1.0	0.0	0.0	0.0	46.0	76.0
4	1.0	1.0	0.0	0.0	0.0	69.0	83.0

Statsmodels

Resulting Design Matrix

```
X.head()
```

	Intercept	Region[T.E]	Region[T.N]	Region[T.S]	Region[T.W]	Literacy	Wealth
0	1.0	1.0	0.0	0.0	0.0	37.0	73.0
1	1.0	0.0	1.0	0.0	0.0	51.0	22.0
2	1.0	0.0	0.0	0.0	0.0	13.0	61.0
3	1.0	1.0	0.0	0.0	0.0	46.0	76.0
4	1.0	1.0	0.0	0.0	0.0	69.0	83.0

Set the Model

```
mod = sm.OLS(y, X)
```

Fit Model

```
res = mod.fit()
```

Statsmodels

Print summary of Model

```
print(res.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                Lottery   R-squared:                       0.338
Model:                            OLS   Adj. R-squared:                  0.287
Method:                 Least Squares   F-statistic:                     6.636
Date:                Mon, 15 Oct 2018   Prob (F-statistic):           1.07e-05
Time:                        22:13:22   Log-Likelihood:                -375.30
No. Observations:                  85   AIC:                             764.6
Df Residuals:                      78   BIC:                             781.7
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      38.6517      9.456      4.087      0.000      19.826      57.478
Region[T.E]   -15.4278      9.727     -1.586      0.117     -34.793       3.938
Region[T.N]   -10.0170      9.260     -1.082      0.283     -28.453       8.419
Region[T.S]    -4.5483      7.279     -0.625      0.534     -19.039       9.943
Region[T.W]   -10.0913      7.196     -1.402      0.165     -24.418       4.235
Literacy       -0.1858      0.210     -0.886      0.378      -0.603       0.232
Wealth          0.4515      0.103      4.390      0.000       0.247       0.656
==============================================================================
Omnibus:                        3.049   Durbin-Watson:                   1.785
Prob(Omnibus):                  0.218   Jarque-Bera (JB):                2.694
Skew:                          -0.340   Prob(JB):                        0.260
Kurtosis:                       2.454   Cond. No.                         371.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

Statsmodels

Print summary of Model

```
print(res.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                Lottery   R-squared:                       0.338
Model:                            OLS   Adj. R-squared:                  0.287
Method:                 Least Squares   F-statistic:                     6.636
Date:                Mon, 15 Oct 2018   Prob (F-statistic):           1.07e-05
Time:                        22:13:22   Log-Likelihood:                -375.30
No. Observations:                  85   AIC:                             764.6
Df Residuals:                      78   BIC:                             781.7
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     38.6517      9.456      4.087      0.000      19.826      57.478
Region[T.E]  -15.4278      9.727     -1.586      0.117     -34.793       3.938
Region[T.N]  -10.0170      9.260     -1.082      0.283     -28.453       8.419
Region[T.S]   -4.5483      7.279     -0.625      0.534     -19.039       9.943
Region[T.W]  -10.0913      7.196     -1.402      0.165     -24.418       4.235
Literacy      -0.1858      0.210     -0.886      0.378      -0.603       0.232
Wealth         0.4515      0.103      4.390      0.000       0.247       0.656
==============================================================================
Omnibus:                        3.049   Durbin-Watson:                   1.785
Prob(Omnibus):                  0.218   Jarque-Bera (JB):                2.694
Skew:                          -0.340   Prob(JB):                        0.260
Kurtosis:                       2.454   Cond. No.                         371.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

Welcome to Statsmodels's Documentation

statsmodels is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration. An extensive list of result statistics are available for each estimator. The results are tested against existing statistical packages to ensure that they are correct. The package is released under the open source Modified BSD (3-clause) license. The online documentation is hosted at statsmodels.org.

Minimal Examples

Since version `0.5.0` of `statsmodels`, you can use R-style formulas together with `pandas` data frames to fit your models. Here is a simple example using ordinary least squares:

```
In [1]: import numpy as np

In [2]: import statsmodels.api as sm

In [3]: import statsmodels.formula.api as smf

# Load data
In [4]: dat = sm.datasets.get_rdataset("Guerry", "HistData").data

# Fit regression model (using the natural log of one of the regressors)
In [5]: results = smf.ols('Lottery ~ Literacy + np.log(Pop1831)', data=dat).fit()

# Inspect the results
In [6]: print(results.summary())
```