

Statistics

Regression

Matthieu Gilson

Statistics in Python

- Python packages:
 - `scipy.stats`
 - `statsmodels`
 - `patsy` (to use R formula)
- Repository: https://etulab.univ-amu.fr/gilson.m/compneuro_course/-/tree/main/stats
 - conda installation
 - environment installation: `environment.yml` file
 - jupyter notebooks
- References (with R language): <https://www.math.univ-toulouse.fr/~besse/Wikistat/>

Course Outline

- Probabilities, distributions
- Parametric and non-parametric testing
- **Regressions**
- Bayesian inference

Regression

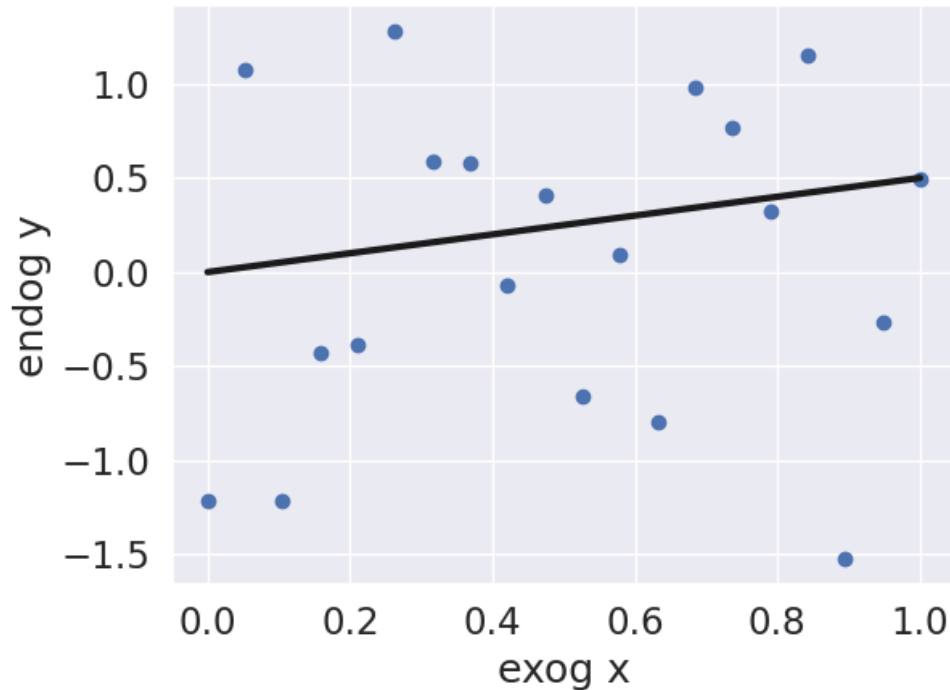
- **Linear regression**
- **Fixed-effect model**
- Mixed-effect model
- Generalized model

Linear Regression

- Simple case of 1 response variable, we check its dependency on the predictor

Linear Regression

- Simple case of 1 response variable, we check its dependency on the predictor



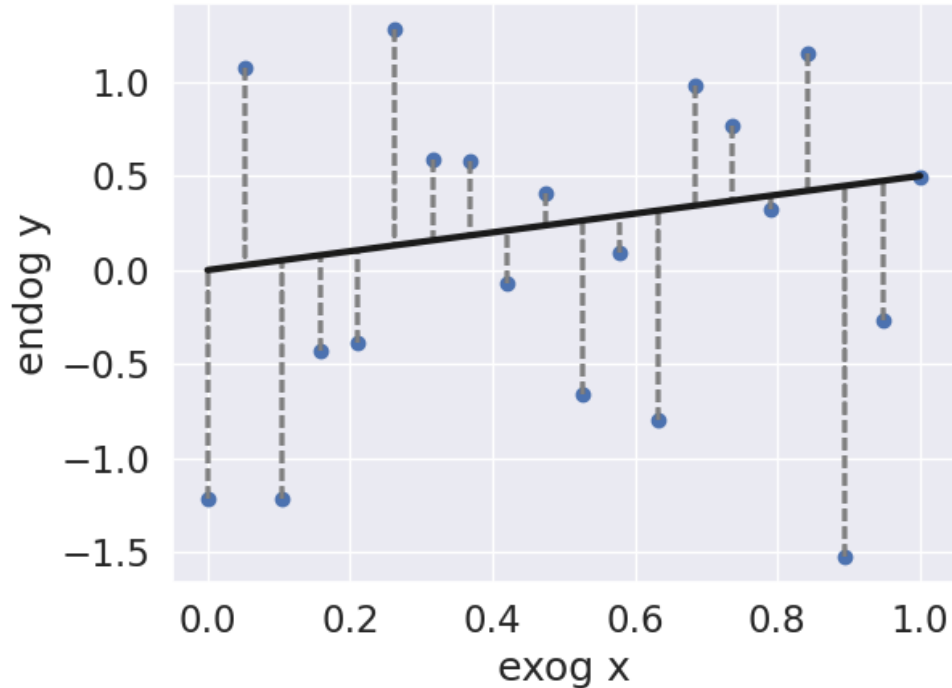
in black: real dependency

$$y = ax + \epsilon$$

normally
distributed
noise

Linear Regression

- Simple case of 1 response variable, we check its dependency on the predictor

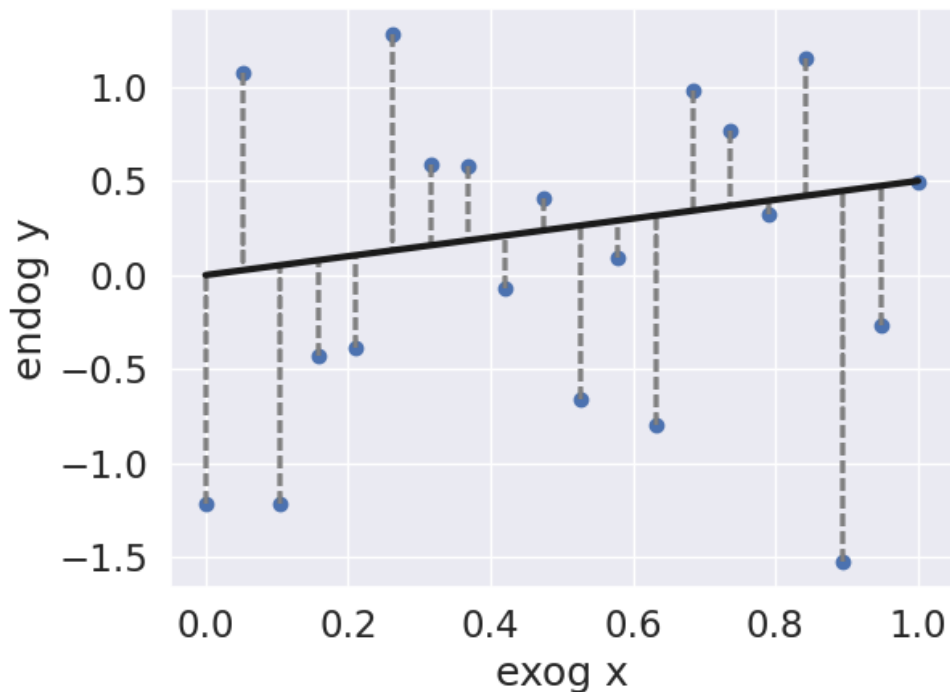


noise in generating samples

$$\epsilon_i = y_i - a x_i$$

Linear Regression

- Simple case of 1 response variable, we check its dependency on the predictor
- Minimize sum of squared errors, assumed to be normally distributed

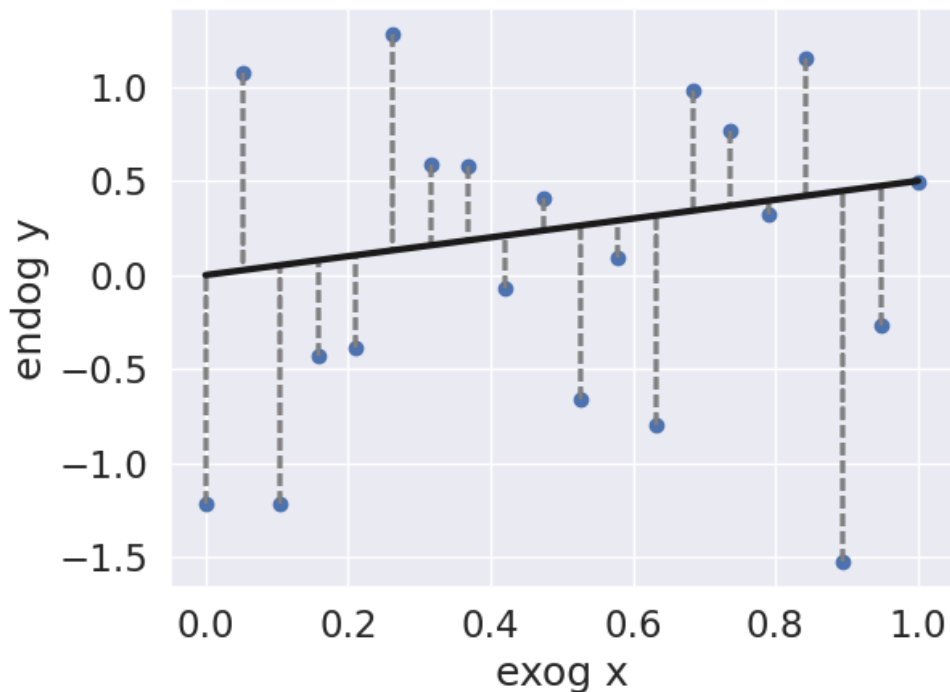


noise in generating samples

$$\epsilon_i = y_i - a x_i$$

Linear Regression

- Simple case of 1 response variable, we check its dependency on the predictor
- Minimize sum of squared errors, assumed to be normally distributed

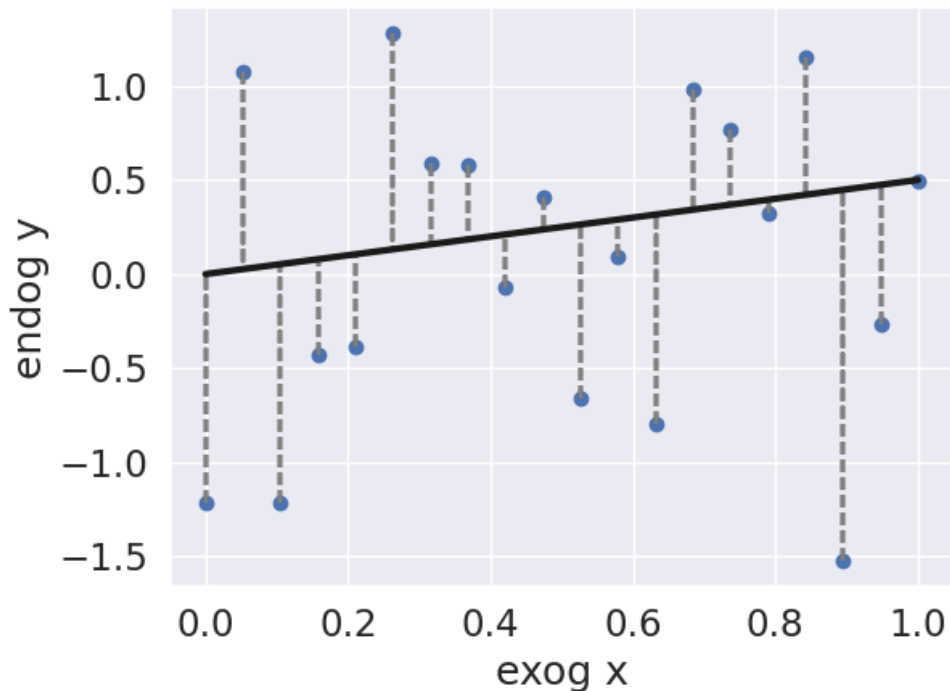


$$\epsilon_i = y_i - a x_i$$

$$\sum_i x_i \epsilon_i = \sum_i x_i y_i - a \sum_i x_i x_i$$

Linear Regression

- Simple case of 1 response variable, we check its dependency on the predictor
- Minimize sum of squared errors, assumed to be normally distributed

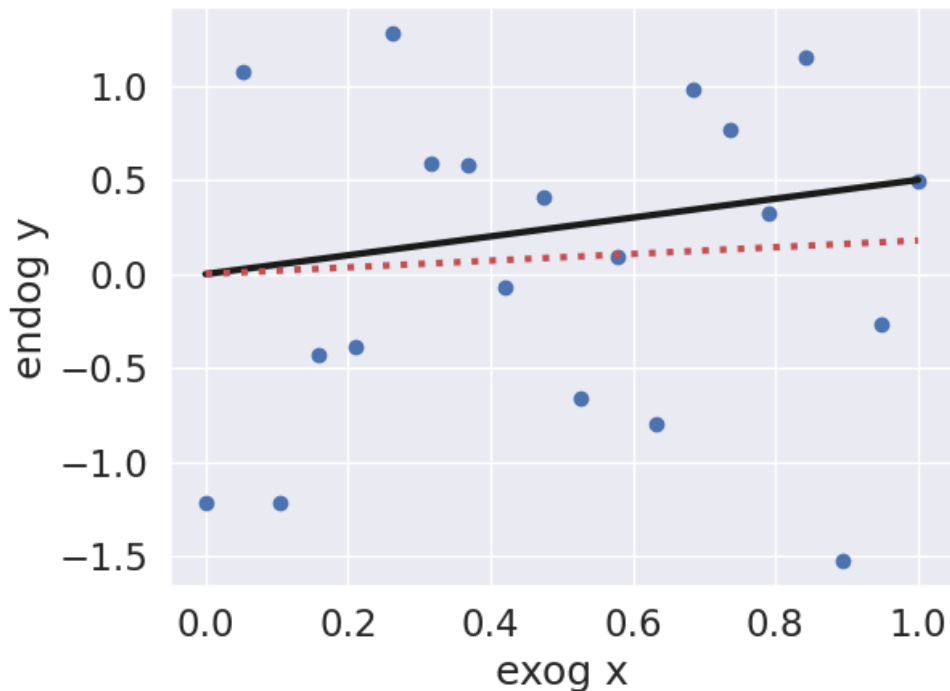


$$\epsilon_i = y_i - a x_i$$

$$0 = \sum_i x_i y_i - a \sum_i x_i x_i$$

Linear Regression

- Simple case of 1 response variable, we check its dependency on the predictor
- Minimize sum of squared errors, assumed to be normally distributed



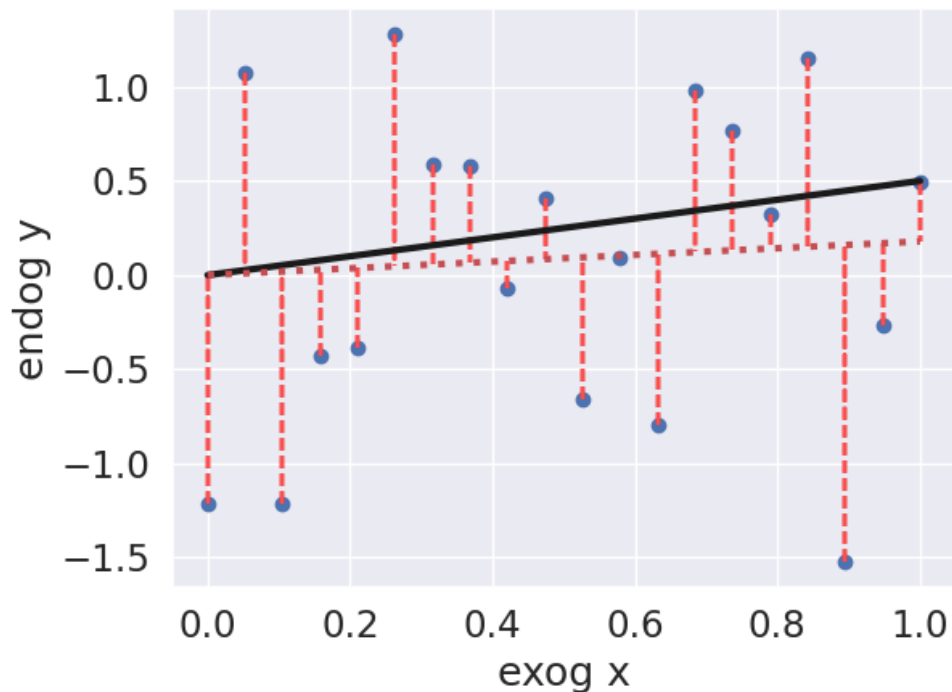
$$\epsilon_i = y_i - a x_i$$

$$\hat{a} = \frac{\sum_i x_i y_i}{\sum_i x_i x_i}$$

estimate
= model
(dotted red)

Linear Regression

- Simple case of 1 response variable, we check its dependency on the predictor
- Minimize sum of squared errors, assumed to be normally distributed



$$\hat{\epsilon}_i = y_i - \hat{a} x_i$$

residuals
= error
(dashed red)

$$\hat{a} = \frac{\sum_i x_i y_i}{\sum_i x_i x_i}$$

estimate
= model
(dotted red)

**DISCREPANCY BETWEEN
ESTIMATE AND TRUE VALUE
DUE TO FINITE NUMBER OF
OBSERVED SAMPLES**

Linear Regression

OLS Regression Results

```
=====
Dep. Variable:          y      R-squared (uncentered):      0.057
Model:                  OLS    Adj. R-squared (uncentered):    0.007
Method:                 Least Squares    F-statistic:          1.144
Date:                  Thu, 20 Jul 2023    Prob (F-statistic):    0.298
Time: 14:47:50    Log-Likelihood:      -25.629
No. Observations:      20    AIC:          53.26
Df Residuals:          19    BIC:          54.25
Df Model:              1
Covariance Type: nonrobust
=====
```

	coef	std err	t	P> t	[0.025	0.975]
x1	0.3656	0.342	1.069	0.298	-0.350	1.081

```
=====
Omnibus:              1.359    Durbin-Watson:      2.199
Prob(Omnibus):        0.507    Jarque-Bera (JB):    0.364
Skew:                 0.283    Prob(JB):            0.834
Kurtosis:             3.340    Cond. No.            1.00
=====
```

Quality and Confidence in Estimation

Quality and Confidence in Estimation

OLS Regression Results

```
=====
Dep. Variable:          y      R-squared (uncentered):      0.069
Model:                  OLS    Adj. R-squared (uncentered):    0.020
Method:                 Least Squares    F-statistic:          1.409
Date:                  Thu, 20 Jul 2023    Prob (F-statistic):    0.250
Time:                  15:40:40    Log-Likelihood:       -30.071
No. Observations:      20      AIC:                  62.14
Df Residuals:          19      BIC:                  63.14
Df Model:               1
Covariance Type:       nonrobust
=====
```

	coef	std err	t	P> t	[0.025	0.975]
x1	0.5067	0.427	1.187	0.250	-0.387	1.400

```
=====
Omnibus:                2.071    Durbin-Watson:          2.495
Prob(Omnibus):          0.355    Jarque-Bera (JB):       0.700
Skew:                   0.358    Prob(JB):               0.705
Kurtosis:               3.573    Cond. No.               1.00
=====
```

Quality and Confidence in Estimation

OLS Regression Results

```
=====
Dep. Variable:          y      R-squared (uncentered):          0.069
Model:                  OLS    Adj. R-squared (uncentered):       0.020
Method:                 Least Squares    F-statistic:          1.409
Date:                  Thu, 20 Jul 2023    Prob (F-statistic):      0.250
Time:                  15:40:40    Log-Likelihood:        -30.071
No. Observations:      20      AIC:                    62.14
Df Residuals:          19      BIC:                    63.14
Df Model:               1
Covariance Type:       nonrobust
=====
```

```
=====
               coef      std err          t      P>|t|      [0.025      0.975]
-----
x1              0.5067      0.427        1.187      0.250      -0.387      1.400
=====
```

```
=====
Omnibus:                2.071    Durbin-Watson:                2.495
Prob(Omnibus):           0.355    Jarque-Bera (JB):             0.700
Skew:                    0.358    Prob(JB):                     0.705
Kurtosis:                3.573    Cond. No.                     1.00
=====
```


Quality and Confidence in Estimation

OLS Regression Results

```
=====
Dep. Variable:          y      R-squared (uncentered):          0.069
Model:                  OLS    Adj. R-squared (uncentered):        0.020
Method:                 Least Squares    F-statistic:          1.409
Date:                  Thu, 20 Jul 2023    Prob (F-statistic):    0.250
Time:                  15:40:40    Log-Likelihood:        -30.071
No. Observations:      20    AIC:                   62.14
Df Residuals:          19    BIC:                   63.14
Df Model:               1
Covariance Type:       nonrobust
=====
```

$$R^2 = \frac{\sum_i (\hat{y}_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2} = 1 - \frac{\sum_i \hat{\epsilon}_i^2}{\sum_i (y_i - \bar{y})^2}$$

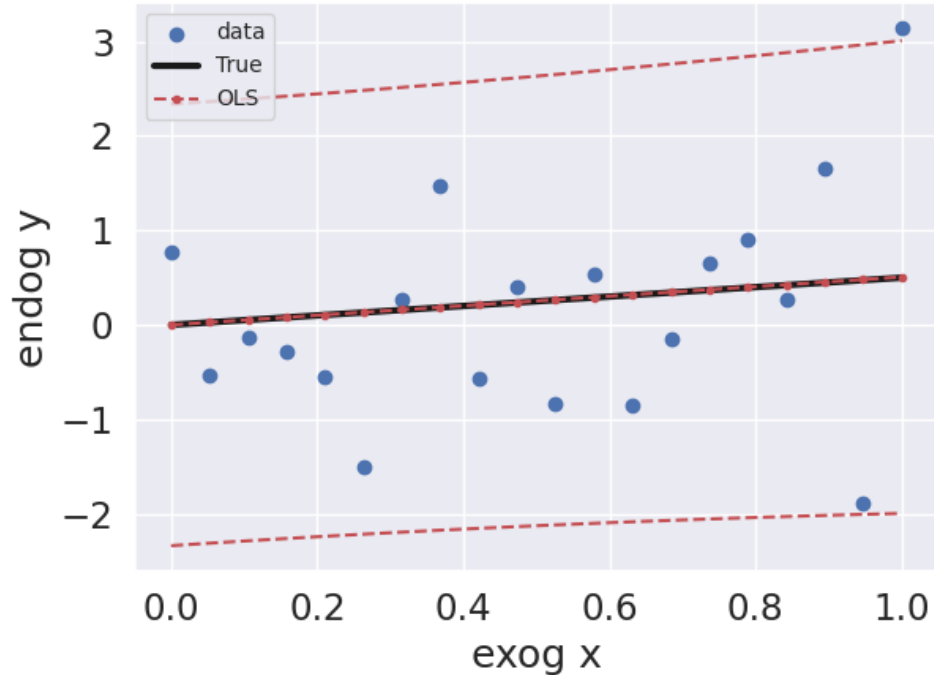
explained variance
total variance

prediction $\hat{y}_i = \hat{a} x_i$

residuals $\hat{\epsilon}_i = y_i - \hat{a} x_i = y_i - \hat{y}_i$

Quality and Confidence in Estimation

$y = a x + \epsilon$ generative model with high noise

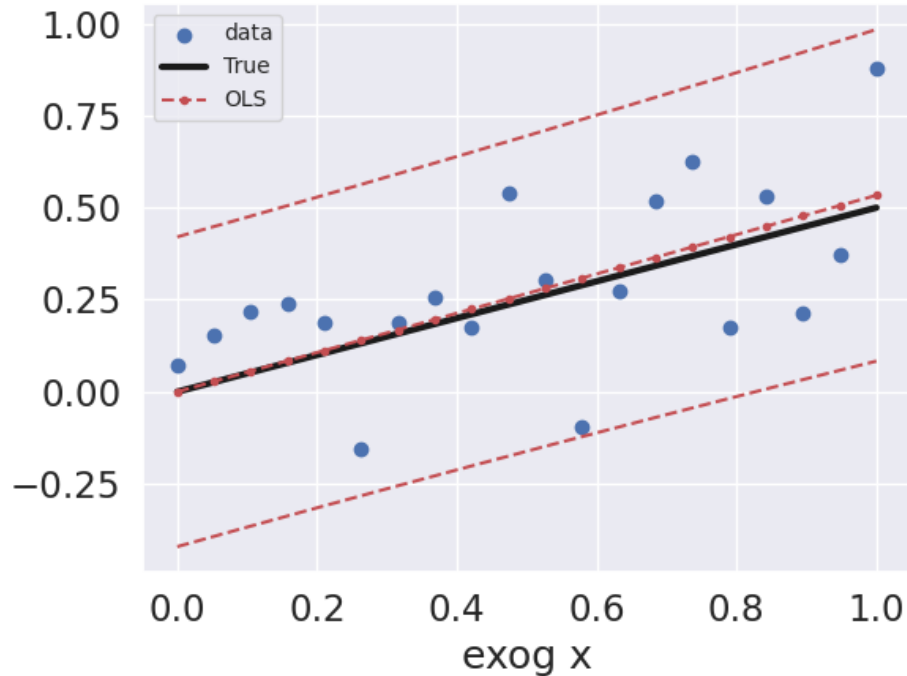


R-squared (uncentered): 0.069
F-statistic: 1.409
Prob (F-statistic): 0.250
Log-Likelihood: -30.071

	coef	std err	[0.025	0.975]
x1	0.5067	0.42	-0.387	1.400

Quality and Confidence in Estimation

$y = a x + \epsilon$ generative model with low noise



R-squared (uncentered): 0.717
F-statistic: 48.25
Prob (F-statistic): 1.28e-06
Log-Likelihood: 4.2313

	coef	std err	[0.025	0.975]
x1	0.5335	0.077	0.373	0.694

Quality and Confidence in Estimation

OLS Regression Results

```
=====
Dep. Variable:          y      R-squared (uncentered):          0.057
Model:                  OLS    Adj. R-squared (uncentered):       0.007
Method:                 Least Squares    F-statistic:          1.144
Date:                  Thu, 20 Jul 2023    Prob (F-statistic):    0.298
Time:                  14:47:50    Log-Likelihood:       -25.629
No. Observations:      20      AIC:          53.26
Df Residuals:          19      BIC:          54.25
Df Model:              1
Covariance Type:       nonrobust
=====
```

	coef	std err	t	P> t	[0.025	0.975]
x1	0.3656	0.342	1.069	0.298	-0.350	1.081

```
=====
Omnibus:              1.359    Durbin-Watson:          2.199
Prob(Omnibus):        0.507    Jarque-Bera (JB):       0.364
Skew:                 0.283    Prob(JB):               0.834
Kurtosis:             3.340    Cond. No.                1.00
=====
```

tests on residuals

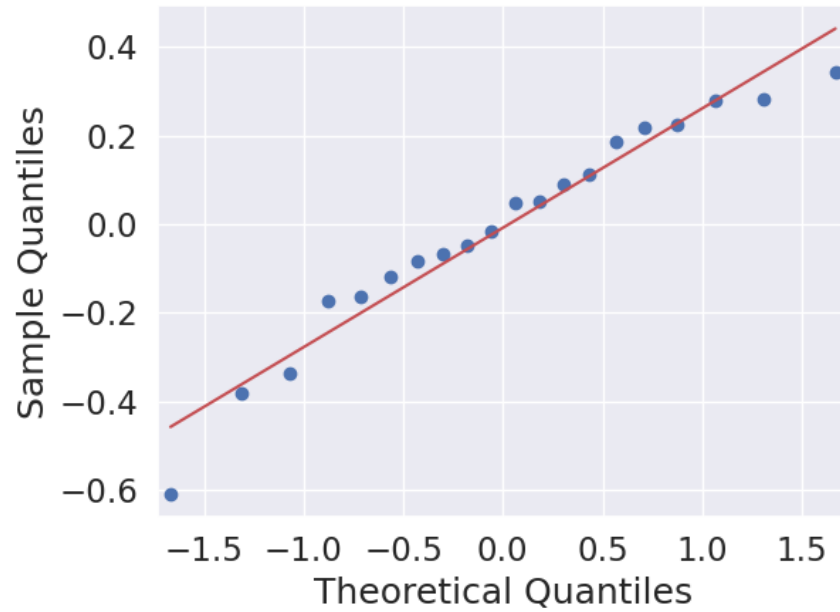
Quality and Confidence in Estimation

- Residuals distributed normally: omnibus test, Jarque-Bera test
- Independent residuals: Durbin-Watson test
 - target value 2 = absence of autocorrelation (< 2 means positive autocorrelation)
 - if statistic between 1 and 3: OK
- Homogeneous residuals (homoscedasticity)

Omnibus:	1.359	Durbin-Watson:	2.199
Prob (Omnibus) :	0.507	Jarque-Bera (JB) :	0.364
Skew:	0.283	Prob (JB) :	0.834
Kurtosis:	3.340	Cond. No.	1.00

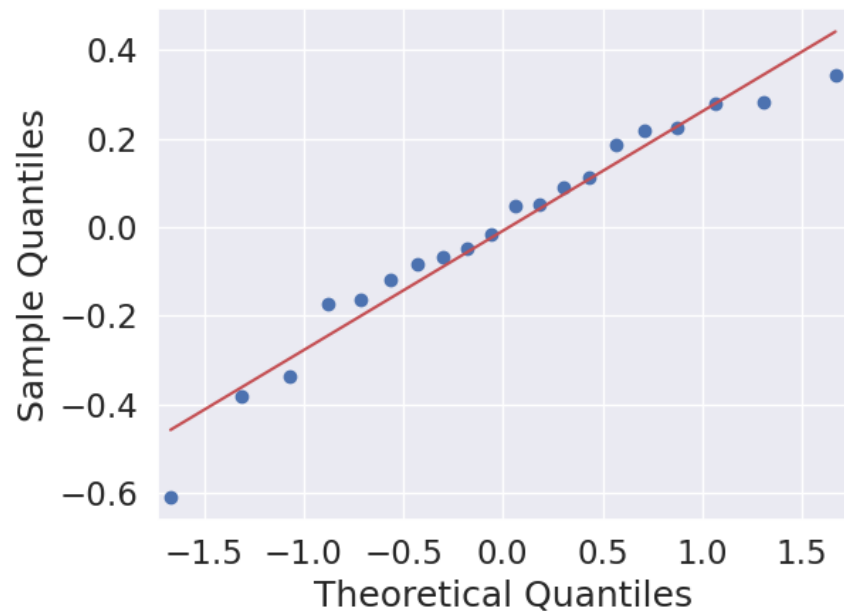
Quality and Confidence in Estimation

normality of residuals: QQ plot

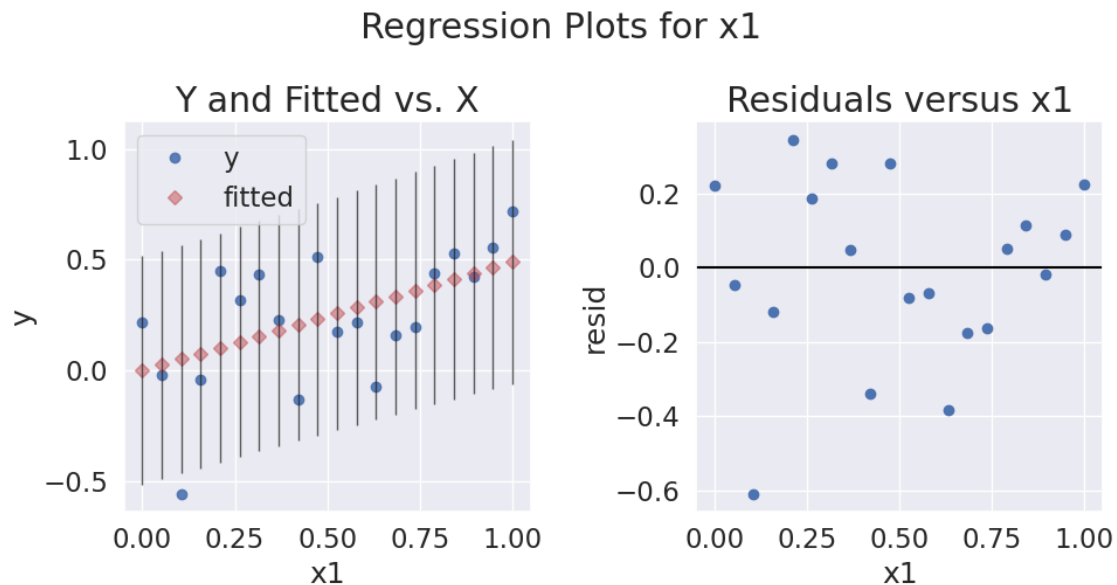


Quality and Confidence in Estimation

normality of residuals: QQ plot



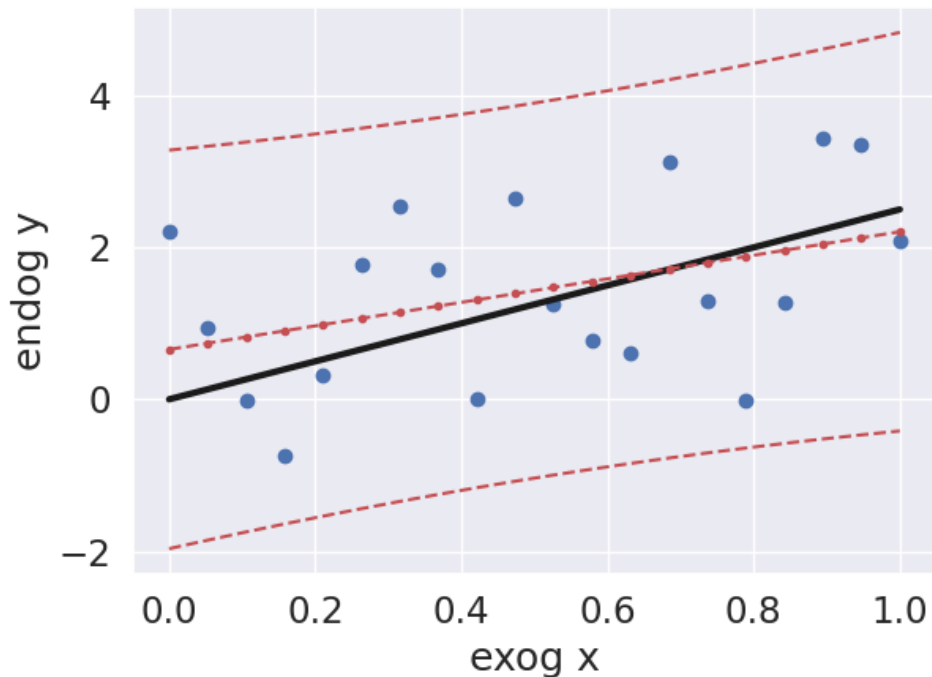
homoscedasticity



Multivariate Linear Regression and Model Comparison

Multivariate Linear Regression and Model Comparison

$y = a x + \epsilon$ linear generative model



- What is the best fitting model for the observed samples?
- If we don't know that they are generated by a linear model, we can test polynomial functions with increasing orders (i.e. increasing complexity)

Multivariate Linear Regression and Model Comparison

OLS Regression Results

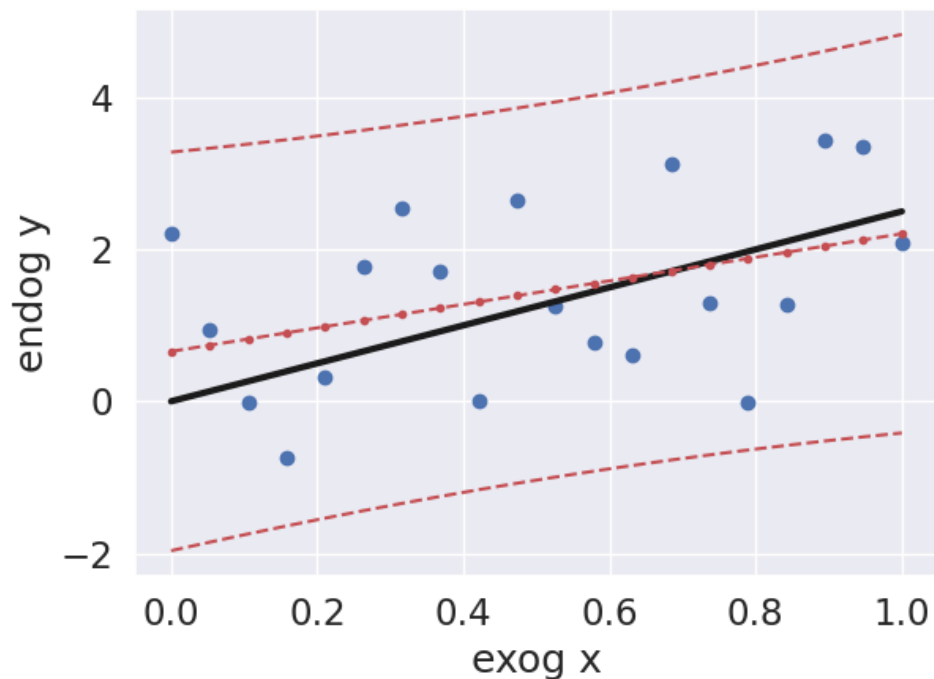
```
=====
Dep. Variable:          y      R-squared:                0.158
Model:                  OLS    Adj. R-squared:           0.111
Method:                 Least Squares    F-statistic:        3.366
Date:                  Tue, 18 Jul 2023    Prob (F-statistic):    0.0831
Time:                  16:37:00    Log-Likelihood:       -30.061
No. Observations:      20    AIC:                 64.12
Df Residuals:          18    BIC:                 66.11
Df Model:               1
Covariance Type:        nonrobust
=====
```

	coef	std err	t	P> t	[0.025	0.975]
const	0.6566	0.494	1.329	0.200	-0.381	1.695
x1	1.5500	0.845	1.835	0.083	-0.225	3.325

```
=====
Omnibus:                4.121    Durbin-Watson:        1.652
Prob(Omnibus):          0.127    Jarque-Bera (JB):     1.412
Skew:                   0.002    Prob(JB):             0.494
Kurtosis:               1.698    Cond. No.:            4.18
=====
```

Multivariate Linear Regression and Model Comparison

polynomial regression $y = \hat{a}_0 + \hat{a}_1 x$

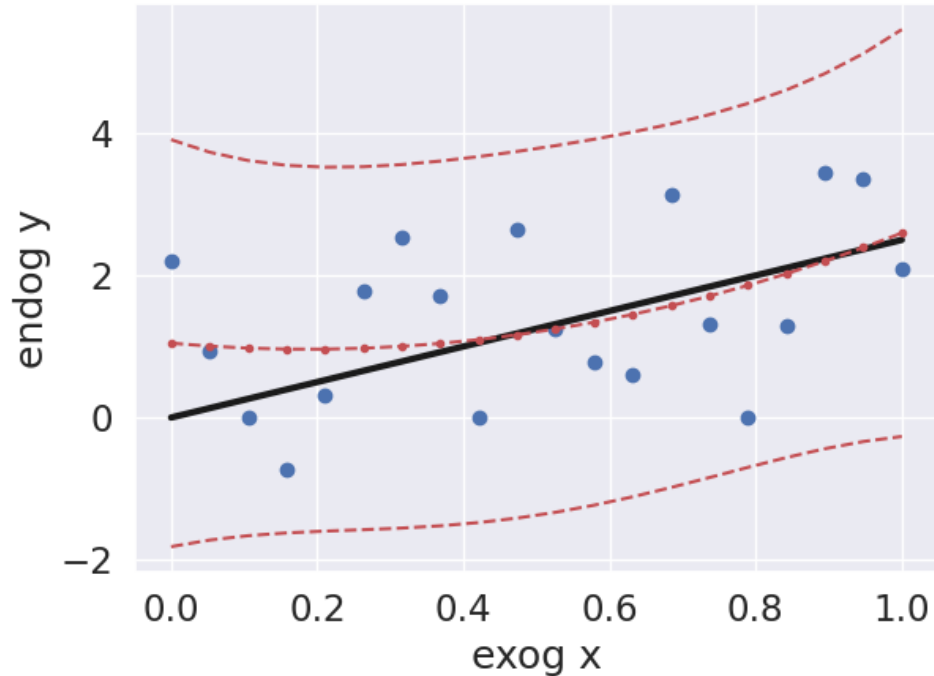


R-squared (uncentered): 0.158
F-statistic: 3.366
Prob (F-statistic): 0.0831
Log-Likelihood: -30.061
BIC: 66.11

	coef	std err	t	P> t
x1	1.5500	0.845	1.835	0.083

Multivariate Linear Regression and Model Comparison

polynomial regression $y = \hat{a}_0 + \hat{a}_1 x + \hat{a}_2 x^2$

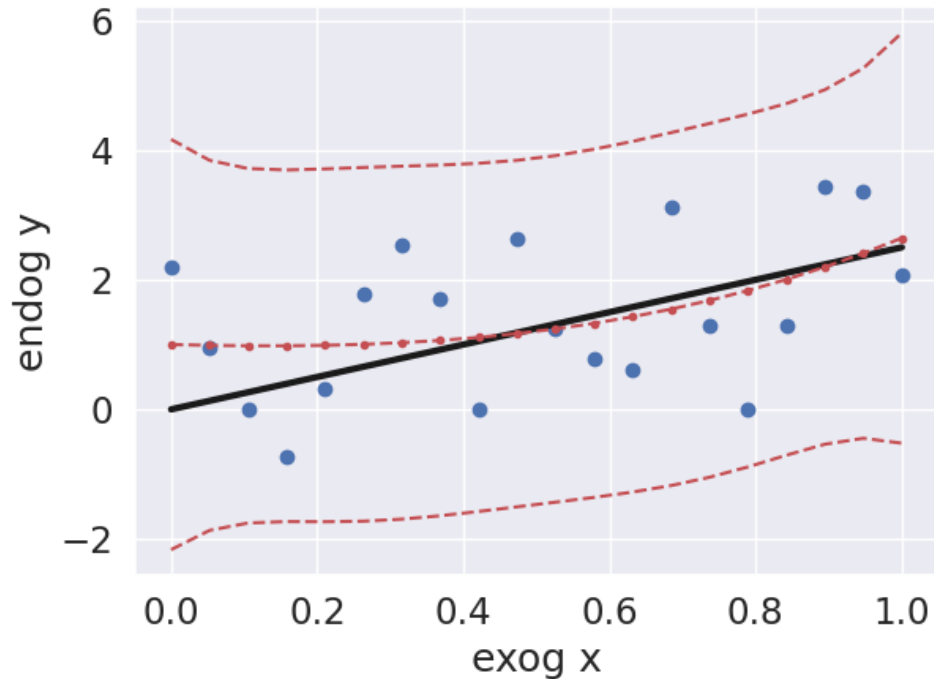


R-squared (uncentered): 0.187
F-statistic: 1.953
Prob (F-statistic): 0.172
Log-Likelihood: -29.707
BIC: 68.40

	coef	std err	t	P> t
x1	-0.9210	3.272	-0.282	0.782
x2	2.4710	3.158	0.782	0.445

Multivariate Linear Regression and Model Comparison

polynomial regression $y = \hat{a}_0 + \hat{a}_1 x + \hat{a}_2 x^2 + \dots$

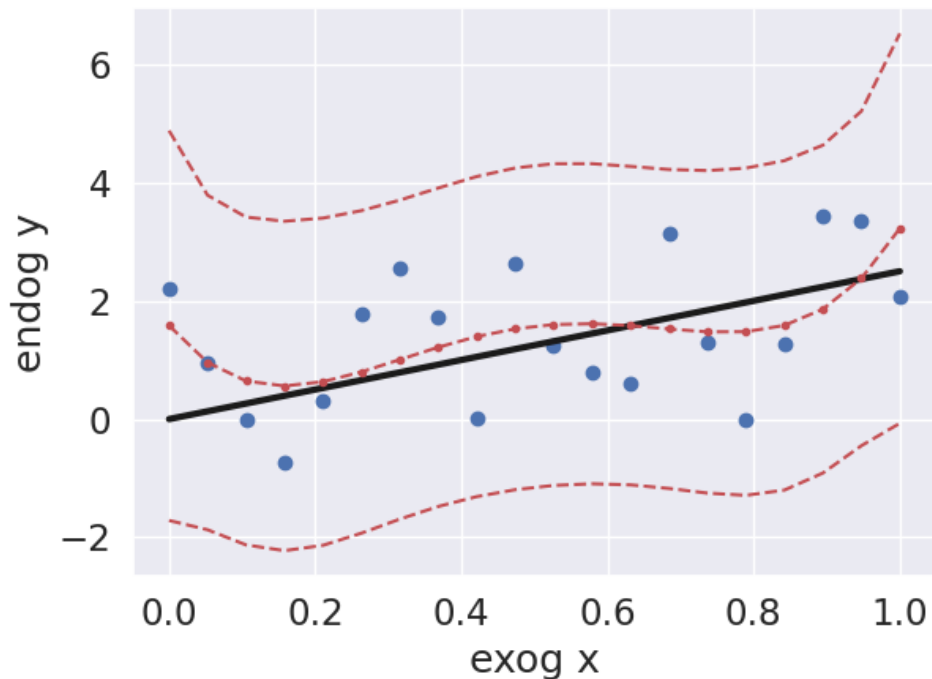


R-squared (uncentered): 0.187
F-statistic: 1.229
Prob (F-statistic): 0.332
Log-Likelihood: -29.702
BIC: 71.39

	coef	std err	t	P> t
x1	-0.2800	7.962	-0.035	0.972
x2	0.8267	18.785	0.044	0.965
x3	1.0963	12.334	0.089	0.930

Multivariate Linear Regression and Model Comparison

polynomial regression $y = \hat{a}_0 + \hat{a}_1 x + \hat{a}_2 x^2 + \dots$

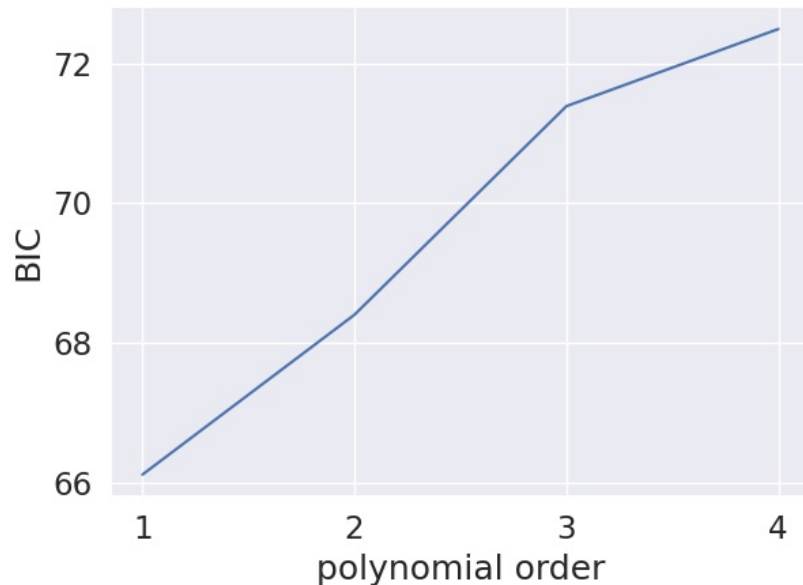
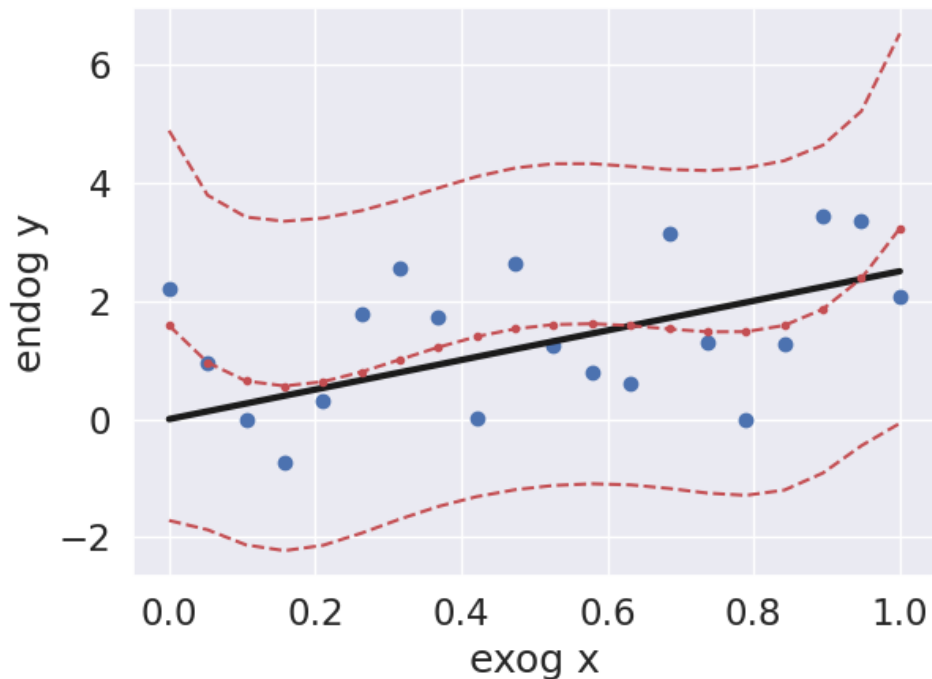


R-squared (uncentered): 0.261
F-statistic: 1.322
Prob (F-statistic): 0.307
Log-Likelihood: -28.757
BIC: 72.49

	coef	std err	t	P> t
x1	-15.4003	14.669	-1.050	0.310
x2	72.9756	61.976	1.177	0.257
x3	-112.9609	94.292	-1.198	0.250
x4	57.0286	46.753	1.220	0.241

Multivariate Linear Regression and Model Comparison

polynomial regression $y = \hat{a}_0 + \hat{a}_1 x + \hat{a}_2 x^2 + \dots$



LOWEST BIC VALUE INDICATES BETTER MODEL
(not R2, nor likelihood, etc.)

Example of Reporting

- Four models have been tested. For i in $\{1,2,3,4\}$, model i corresponds to regressors $\{x^0, \dots, x^i\}$ that are powers of x . We find that model 1 is the best with the lowest BIC/AIC value.
- The best model is at the border of significance with $F(1, 18) = 3.366$, $p = 0.08$, explaining 15,8% of the variance of y .
 - in the parentheses of F , the degrees of freedoms (“Df” in the output table of `scipy.stats`) correspond to the model and to the data
 - for the model, we have $1 = 2 - 1$ for the polynomial of order 1: it is equal the number of parameters (2, the coefficient and the intercept) minus 1
 - for the data, we have $18 = 20 - 2$, the number of time points minus the number of model parameters

Building a Model with Formula

Building a Model with Formula

- The *patsy* package facilitates the construction of design matrices using R's formula
- The models can be built using *statsmodels.formula.api*

dataframe

	x1	x2	y
0	0.226948	0.358764	0.847936
1	0.531380	0.930131	-0.908091
2	1.484127	1.642451	0.428391
3	0.363713	0.023218	0.355144
4	-0.032484	-0.006205	0.577427

$$y = a_1 x_1 + a_2 x_2 + \epsilon$$

Building a Model with Formula

- The *patsy* package facilitates the construction of design matrices using R's formula
- The models can be built using *statsmodels.formula.api*

dataframe

	x1	x2	y
0	0.226948	0.358764	0.847936
1	0.531380	0.930131	-0.908091
2	1.484127	1.642451	0.428391
3	0.363713	0.023218	0.355144
4	-0.032484	-0.006205	0.577427

'y ~ x1 + x2'

design matrix

	Intercept	x1	x2
0	1.0	0.226948	0.358764
1	1.0	0.531380	0.930131
2	1.0	1.484127	1.642451
3	1.0	0.363713	0.023218
4	1.0	-0.032484	-0.006205

Building a Model with Formula

true values

```
=====
Dep. Variable:                y      R-squared:                0.484
Model:                        OLS     Adj. R-squared:           0.462
Method:                      Least Squares   F-statistic:             22.01
Date:                        Thu, 06 Jul 2023   Prob (F-statistic):      1.80e-07
Time:                        00:36:28     Log-Likelihood:         -62.306
No. Observations:            50      AIC:                    130.6
Df Residuals:                47      BIC:                    136.3
Df Model:                    2
Covariance Type:             nonrobust
=====
               coef      std err          t      P>|t|      [0.025      0.975]
-----
0.0 Intercept      -0.0413      0.125      -0.332      0.741      -0.292      0.209
-0.5 x1            -0.4097      0.118      -3.468      0.001      -0.647     -0.172
0.7 x2             0.5906      0.136       4.335      0.000       0.316      0.865
=====
Omnibus:                1.032   Durbin-Watson:           2.003
Prob(Omnibus):           0.597   Jarque-Bera (JB):         0.986
Skew:                   0.167   Prob(JB):                 0.611
Kurtosis:               2.398   Cond. No.                 1.47
=====
```

Building a Model with Formula

- Now let's consider another target y_2 with a multiplicative interaction from x_1 and x_2 , in addition to the linear dependencies

dataframe

	x1	x2	y	y2
0	-0.598903	-1.107940	-0.983083	-0.717664
1	1.484895	-0.308404	-2.069647	-2.252826
2	-1.279121	-0.071924	0.927597	0.964397
3	0.796557	-1.757192	-0.702349	-1.262231
4	0.772527	0.049753	0.529516	0.544891

$$y_2 = a_1 x_1 + a_2 x_2 + a_{12} x_1 x_2 + \epsilon$$

Building a Model with Formula

- Now let's consider another target y_2 with a multiplicative interaction from x_1 and x_2 , in addition to the linear dependencies
- The formula below (in red) only considers linear dependencies

dataframe

	x1	x2	y	y2
0	-0.598903	-1.107940	-0.983083	-0.717664
1	1.484895	-0.308404	-2.069647	-2.252826
2	-1.279121	-0.071924	0.927597	0.964397
3	0.796557	-1.757192	-0.702349	-1.262231
4	0.772527	0.049753	0.529516	0.544891

'y ~ x1 + x2'

design matrix

	Intercept	x1	x2
0	1.0	0.226948	0.358764
1	1.0	0.531380	0.930131
2	1.0	1.484127	1.642451
3	1.0	0.363713	0.023218
4	1.0	-0.032484	-0.006205

$$y_2 = a_1 x_1 + a_2 x_2 + a_{12} x_1 x_2 + \epsilon$$

Building a Model with Formula

true values

0.0
-0.5
0.7

```
=====
Dep. Variable:                y2      R-squared:                0.347
Model:                        OLS      Adj. R-squared:           0.319
Method:                       Least Squares      F-statistic:           12.48
Date:                         Thu, 06 Jul 2023      Prob (F-statistic):      4.48e-05
Time:                         00:36:52      Log-Likelihood:         -67.852
No. Observations:             50      AIC:                    141.7
Df Residuals:                 47      BIC:                    147.4
Df Model:                     2
Covariance Type:              nonrobust
=====
               coef      std err          t      P>|t|      [0.025      0.975]
-----
Intercept    -0.1561      0.139      -1.121      0.268     -0.436      0.124
x1           -0.3311      0.132      -2.509      0.016     -0.597     -0.066
x2            0.5106      0.152       3.354      0.002      0.204      0.817
=====
Omnibus:            0.529      Durbin-Watson:           2.231
Prob(Omnibus):      0.768      Jarque-Bera (JB):         0.634
Skew:               -0.016      Prob(JB):                 0.728
Kurtosis:           2.449      Cond. No.                 1.47
=====
```

Building a Model with Formula

- To consider a multiplicative interaction dependency, we need to adjust the formula (“+” replaced by “*”)
- Note that it also involves linear dependencies (columns in the design matrix)

dataframe

	x1	x2	y	y2
0	-0.598903	-1.107940	-0.983083	-0.717664
1	1.484895	-0.308404	-2.069647	-2.252826
2	-1.279121	-0.071924	0.927597	0.964397
3	0.796557	-1.757192	-0.702349	-1.262231
4	0.772527	0.049753	0.529516	0.544891

'y ~ x1 * x2'

design matrix

	Intercept	x1	x2	x1:x2
0	1.0	-0.598903	-1.107940	0.663548
1	1.0	1.484895	-0.308404	-0.457947
2	1.0	-1.279121	-0.071924	0.092000
3	1.0	0.796557	-1.757192	-1.399704
4	1.0	0.772527	0.049753	0.038436

Building a Model with Formula

```
=====
Dep. Variable:                y2      R-squared:                0.477
Model:                        OLS      Adj. R-squared:           0.443
Method:                      Least Squares  F-statistic:             14.01
Date:                        Thu, 06 Jul 2023  Prob (F-statistic):      1.28e-06
Time:                        00:36:41      Log-Likelihood:          -62.280
No. Observations:            50      AIC: 132.6
Df Residuals:                46      BIC: 140.2
Df Model:                    3
Covariance Type:              nonrobust
=====
```

true values

0.0
-0.5
0.7
0.4

```
=====
              coef      std err          t      P>|t|      [0.025      0.975]
-----
Intercept    -0.0335      0.131      -0.256      0.799      -0.297      0.230
x1           -0.4150      0.122     -3.405      0.001      -0.660     -0.170
x2            0.5960      0.140      4.259      0.000      0.314      0.878
x1:x2         0.4273      0.126      3.389      0.001      0.174      0.681
=====
```

```
=====
Omnibus:            0.909      Durbin-Watson:           1.989
Prob(Omnibus):      0.635      Jarque-Bera (JB):         0.917
Skew:               0.164      Prob(JB):                 0.632
Kurtosis:           2.423      Cond. No.                  1.71
=====
```

Building a Model with Formula

MORE COMPLEX MODEL, BUT BETTER THAN WITHOUT INTERACTION!

```
=====
Dep. Variable:          y2      R-squared:                0.477
Model:                  OLS     Adj. R-squared:           0.443
Method:                 Least Squares   F-statistic:         14.01
Date:                  Thu, 06 Jul 2023   Prob (F-statistic):    1.28e-06
Time:                  00:36:41   Log-Likelihood:       -62.280
No. Observations:      50      AIC: 132.6
Df Residuals:          46      BIC: 140.2
Df Model:              3
Covariance Type:       nonrobust
=====
```

true values

0.0
-0.5
0.7
0.4

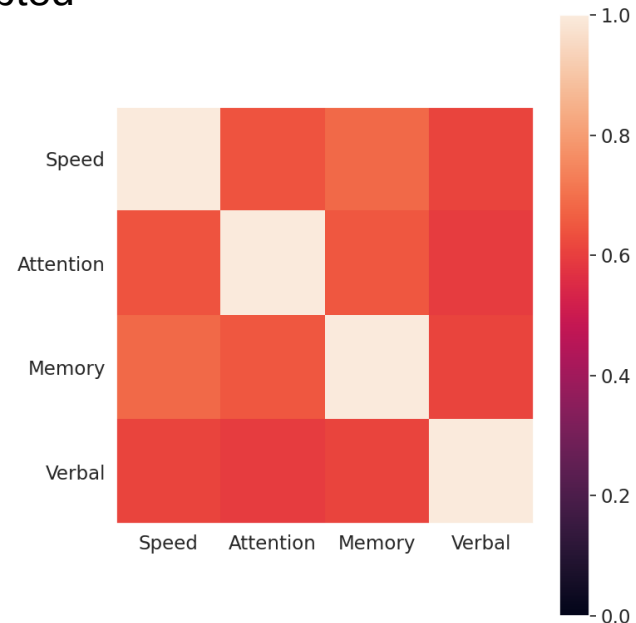
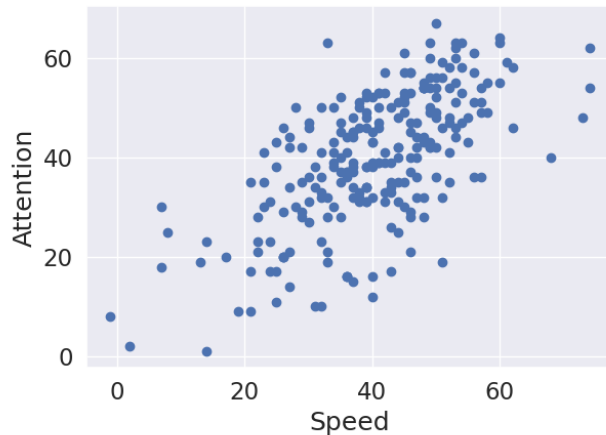
```
=====
              coef      std err          t      P>|t|      [0.025      0.975]
-----
Intercept    -0.0335      0.131      -0.256      0.799      -0.297      0.230
x1           -0.4150      0.122     -3.405      0.001     -0.660     -0.170
x2            0.5960      0.140      4.259      0.000      0.314      0.878
x1:x2         0.4273      0.126      3.389      0.001      0.174      0.681
=====
```

```
=====
Omnibus:      0.909      Durbin-Watson:      1.989
Prob(Omnibus): 0.635      Jarque-Bera (JB):      0.917
Skew:         0.164      Prob(JB):      0.632
Kurtosis:     2.423      Cond. No.      1.71
=====
```

Multivariate Linear Regression

Multivariate Linear Regression

- In addition to the conditions in the quality check (residuals, etc.), a multivariate regression with several predictor variables should involve predictors that are not too correlated across all data.
- In practice, this can simply be checked via a correlation matrix between all of them
 - values below 0,8 or even 0,9 are typically accepted



Reporting for Multivariate Linear Regression

- The tested model is significant, $F(3, 238) = 103.8$, $p < 1e-10$, adjusted- $R^2 = 0.56$. The coefficient for Attention is significant ($\beta = 0.24$, $p < 1e-10$), as well as Memory ($\beta = 0.37$, $p < 1e-10$) and Verbal ($\beta = 0.26$, $p < 1e-10$).
 - note that the adjusted R^2 is often reported, instead of the plain one; this is because R^2 tends to increase when more predictors are used
 - confidence intervals can be reported especially to stress that the coefficient is estimated to be positive/negative

Dep. Variable:	Speed	R-squared:	0.567
Model:	OLS	Adj. R-squared:	0.561
No. Observations:	242	F-statistic:	103.8
Df Residuals:	238	Prob (F-statistic):	5.52e-43
Df Model:	3	Log-Likelihood:	-845.12

	coef	std err	t	P> t	[0.025	0.975]
Intercept	4.0482	2.251	1.798	0.073	-0.386	8.483
Attention	0.2393	0.053	4.496	0.000	0.134	0.344
Memory	0.3726	0.060	6.242	0.000	0.255	0.490
Verbal	0.2643	0.068	3.897	0.000	0.131	0.398

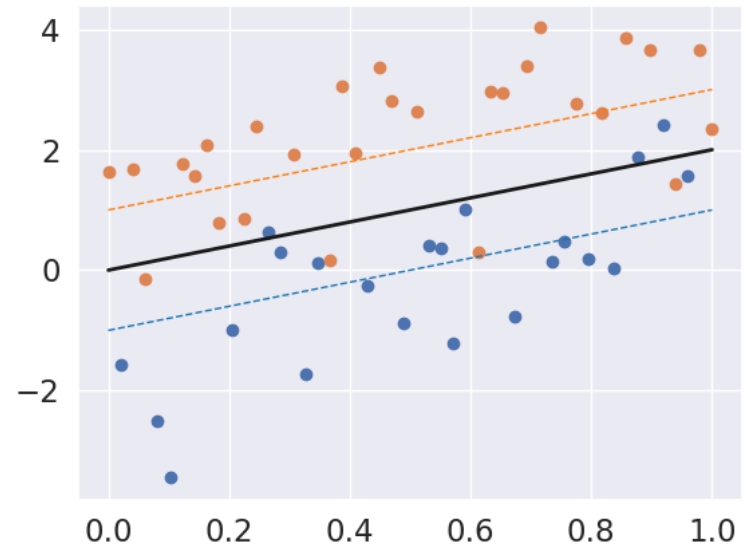
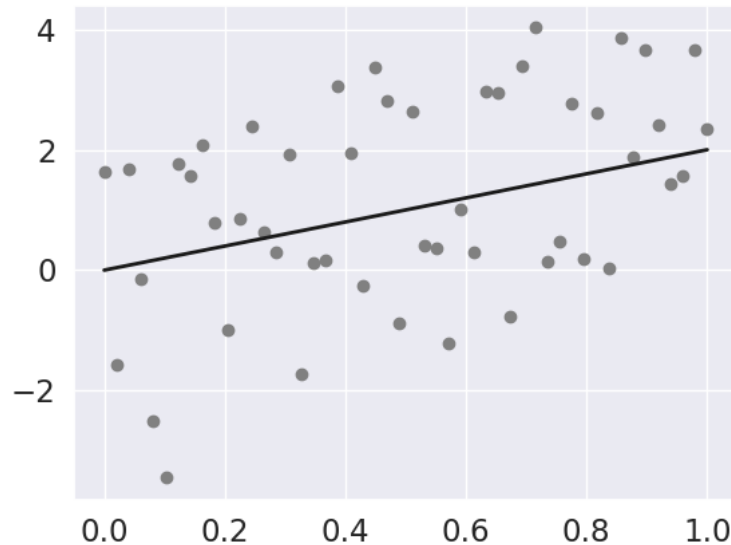
Statistical Analysis in Python

- Probabilities, distributions
- Parametric and non-parametric testing
- **Regressions**
 - **linear regression**
 - fixed-effect model
 - **mixed-effect model**
- Bayesian inference

Mixed Model

Mixed Model

- Example of distinct baselines for 2 sample groups: we know that the samples are in fact comprising of 2 groups, each with a distinct baseline (intercept)



Practice

notebook DESU_regression

Statistical Analysis in Python

- Probabilities, distributions
- Parametric and non-parametric testing
- **Regressions**
 - **categorical predictor variables**
 - **ANOVA**
- Bayesian inference

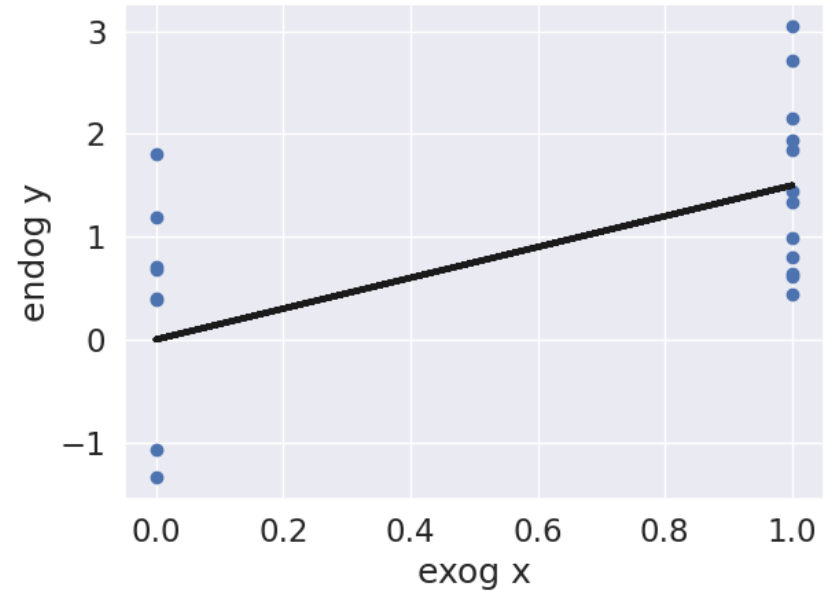
Linear Regression with Categorical Variables

- Back to simple case
- Same number of samples in each group (n)

$$y = ax + \epsilon$$

$$x \in \{0, 1\}$$

group index



Linear Regression with Categorical Variables

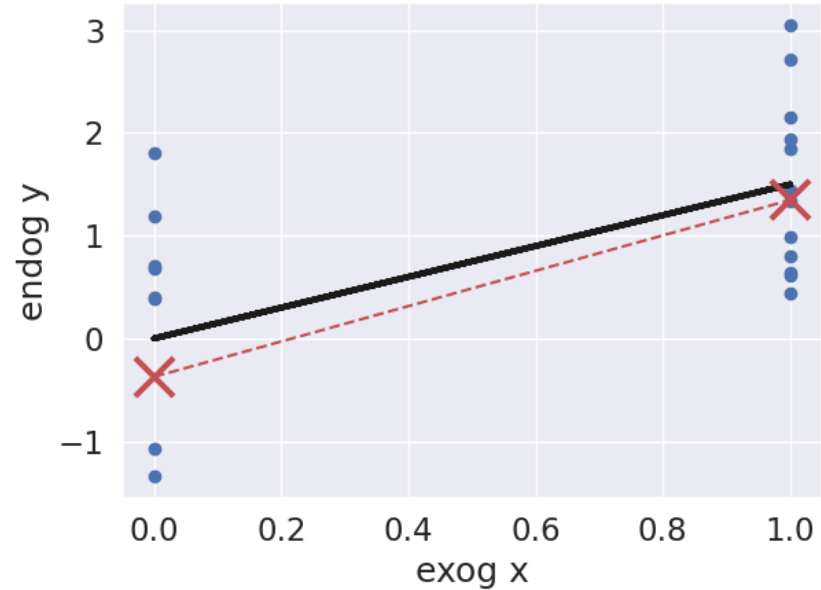
- Back to simple case
- Same number of samples in each group (n)

$$y = a x + \epsilon$$

$x \in \{0, 1\}$
group index

$$y_i = a x_i + \epsilon_i \quad \longrightarrow \quad \hat{a} = \bar{y}_1 - \bar{y}_0$$

\bar{y}_x average of group x



Linear Regression with Categorical Variables

- Back to simple case
- Same number of samples in each group (n)

$$y = a x + \epsilon$$

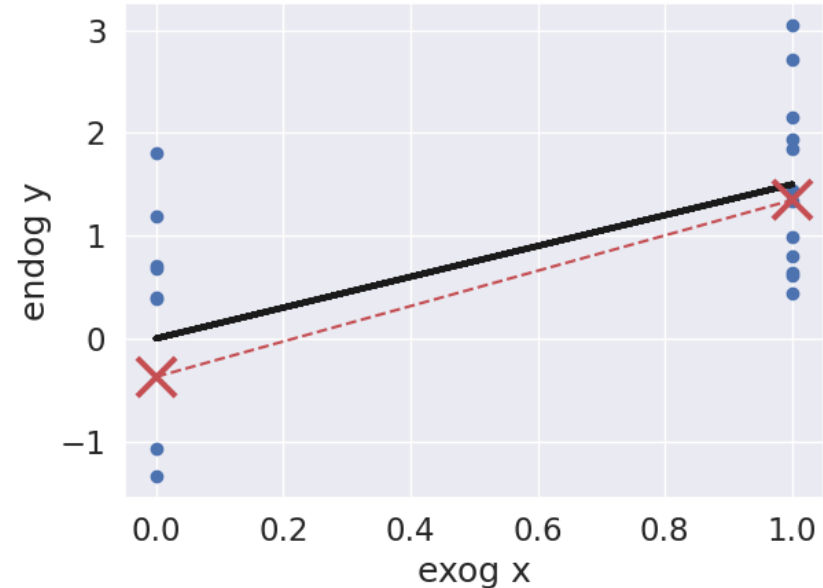
$$x \in \{0, 1\}$$

group index

$$y_i = a x_i + \epsilon_i \quad \longrightarrow \quad \hat{a} = \bar{y}_1 - \bar{y}_0$$

\bar{y}_x average of group x

**ESTIMATE REFLECTS MEAN
DIFFERENCE BETWEEN GROUPS**



Linear Regression with Categorical Variables

- Back to simple case
- Same number of samples in each group (n)

$$y = a x + \epsilon$$

$$x \in \{0, 1\}$$

group index

t statistic for significance
(slope estimate non zero)

$$t = \frac{\hat{a} - 0}{s.e.} = \frac{\bar{y}_1 - \bar{y}_0}{\sigma_{est} / \sqrt{n}}$$

same as Student's t test with equal variance...

s.e. = standard error of estimate
(function of variance of estimate)

$$\sigma_{est} = var(\bar{y}_0) + var(\bar{y}_1)$$

$$= \frac{1}{n-1} \sum_{i \in x} (y_i - \bar{y}_x)^2$$

Linear Regression with Categorical Variables

- Back to simple case
- Same number of samples in each group (n)

$$y = a x + \epsilon$$

$$x \in \{0, 1\}$$

group index

F statistic for ANOVA:

$$F = \frac{\sum_x n (\bar{y}_x - \bar{y})^2 / 1}{\sum_{x,i} (y_i - \bar{y}_x)^2 / (2n - 2)}$$

BSS

WSS

with $\bar{y} = \frac{\bar{y}_1 + \bar{y}_0}{2}$

- explained sum of squares (between-group sum of squares, BSS)
- unexplained sum of square (within-group sum of squares, WSS)
- total sum of squares (TSS = BSS + WSS)
- note degrees of freedom

Linear Regression with Categorical Variables

- Back to simple case $y = a x + \epsilon$ $x \in \{0, 1\}$
- Same number of samples in each group (n) group index
- In fact, we have $F = t^2$ and the two tests give the same p-value!
 - BSS relates to \hat{a}
 - WSS relates to σ_{est}

Typology of Statistical Tests

- Sum of squares of normal random variables (e.g. residuals)
 - chi-square distribution (cf. degrees of freedom)

$$\sum_i \epsilon_i^2$$

- Ratio of sum of squares (e.g. explained versus unexplained variance)
 - F statistic

$$\frac{\sum_i \epsilon_i^2}{\sum_i \xi_i^2}$$

- Ratio of estimate by its variability (standard error) to test if
 - t statistic (and distribution)

$$\frac{\hat{a}}{s.e.(\hat{a})} = \frac{\hat{a}}{\sigma_{\hat{a}}/\sqrt{n}}$$