Universität Zürich[UZH]

Soziologisches Institut

Data Analysis – Advanced Statistics with Python

Dr. Julia Jerke

jerke@soziologie.uzh.ch

Thursday, 12.15pm – 13.45pm, AND 2.46

Session 4 – Logistic regression

Agenda

1. Dummy variables and interaction terms in regression models
2. Logistic regression basics
3. Logistic regression with *Python*
4. Hands on

1. Dummy variables and interaction terms in regression models

Motivation

- The standard case of linear regression is the relationship between a ***continuous*** dependent variable Y and one or more ***continuous*** explanatory variables $X_1, X_2, X_3, \ldots$
- But what if one or more of the variables are not continuous?

$\boldsymbol{Y}$ / $\boldsymbol{X_1, X_2, X_3, \ldots}$	**Continuous**	**Binary (categorical)**
Continuous	Linear regression	Logistic regression (Logit), Ordinal regression
Binary (categorical)	Dummy and interaction terms	Logit with dummy and interaction, Chi square analyses

Linear regression with dummy variables

- In the classic case, the dependent and independent variables are continuous
- However, the independent variables can also be dummy or categorical variables
  - ➢ In the example of the test performance and the preparation time we could, for instance, also include the effect of the gender (*)
- Assume we have a binary variable D
- The interpretation of the coefficient for the dummy variable can be expressed as follows:

$\hat{y} = b_0 + b_1 * x + b_2 * d$	$d = 0$	$d = 1$
$\hat{y} =$	$b_0 + b_1 * x$	$(b_0 + b_2) + b_1 * x$

➢ b_2 is the expected difference in Y between the two values of D (ceteris paribus)

(*) This serves just as a simplistic example. Today, gender is rightfully no longer asked in a binary way in most surveys.

Linear regression with interaction variables

- What if the relationship between Y and X differs for different groups‘
  - In the example of the test performance and the preparation time we could, for instance, also analyse whether the effect of the preparation time is different for female and male students
- We can include an interaction term $\mathrm{X} * D$ into the regression model
- The interpretation of the coefficient for the dummy variable can be expressed as follows:

$\hat{y} = b_0 + b_1 * x + b_2 * d + b_3 * x \cdot d$	$d = 0$	$d = 1$
$\hat{y} =$	$b_0 + b_1 * x$	$(b_0 + b_2) + (b_1 + b_3) * x$

  - b_3 is the expected difference in the effect of X on Y between the two values of D (ceteris paribus)
  - b_2 is the expected difference in the intercept between the two values of D

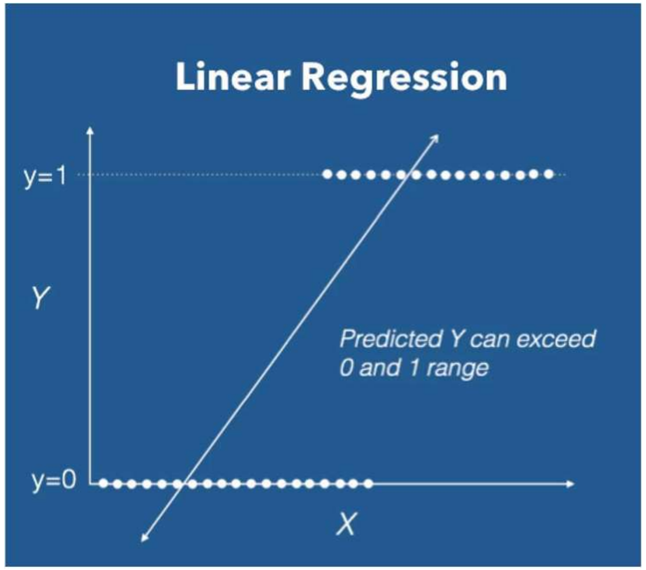
2. Logistic regression basics

Logistic versus linear regression

- Specifying a (non-)linear relationship between a **binary** dependent variable Y and one or more explanatory variables $X_1, X_2, X_3, \ldots$
  - *Simple logistic regression:* one explanatory variable X_1
  - *Multiple logistic regression:* two or more explanatory variables $X_1, X_2, \ldots, X_p$:
- In the case of logistic regression, we usually want to predict the probability that an observation either belongs to $Y = 1$ or $Y = 0$
- Linear regression is not appropriate anymore:
  - The dependent variable Y now follows a binomial distribution which poses a problem for the linear regression
  - The classic linear regression model

$$\mathrm{y_i} = \beta_0 + \beta_1 * \mathrm{x_{i1}} + \beta_2 * \mathrm{x_{i2}} + \cdots + \beta_\mathrm{p} * \mathrm{x_{ip}} + \epsilon_\mathrm{i}$$

cannot be used since it makes continuous predictions



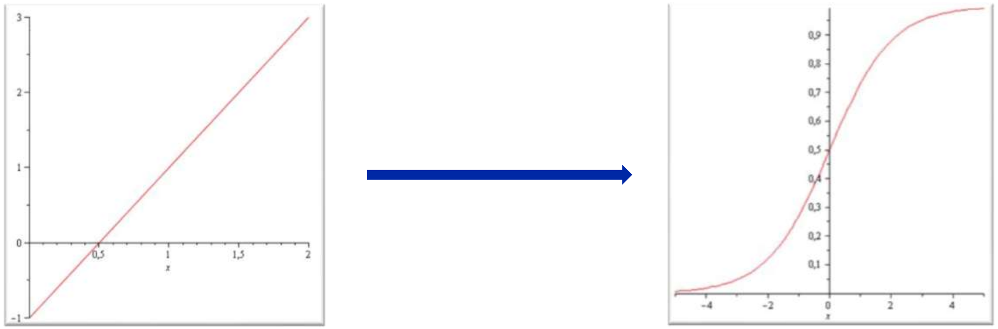
https://www.datacamp.com/community/tutorials/logistic-regression-R

From linear to logistic regression I

- **Goal**: Modelling the probability of belonging to $Y = 1$ instead of $Y = 0$
- **Solution**: Transforming the classic regression equation to a regression equation with a range of $[0,1]$
- **Procedure:**
  (1) Start with the regression equation: $\hat{y} = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_p * x_p$

  (2) Transformation with a link function h : $h(\hat{y}) = h(\beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_p * x_p)$

  (3) A good choice for h is the logistic function $h(y) = \frac{1}{1+e^{-y}}$



  (4) Since the logistic function is a probability distribution, we have: $h(y) = P(y = 1)$ and therefore:

$$P(y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_p * x_p)}}$$

From linear to logistic regression II

- We now have a transformed regression equation:

$$P(y=1) = \frac{1}{1+e^{-(\beta_0+\beta_1*x_1+\beta_2*x_2+\cdots+\beta_p*x_p)}}$$

- In this way, for any person in the dataset, the probability of belonging to $Y=1$ given their values for $X_1, X_2, X_3, \ldots$ can be predicted
- *But*: how can we estimate the specific influence of $X_1, X_2, X_3, \ldots$, in the sense of. "If X_j increases by one unit, then ..."?
- Rearranging and adding the error term yields the final regression equation:

$$\ln\left(\frac{P(y=1)}{1-P(y=1)}\right) = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_p * x_p + \epsilon_i, \qquad i = 1, \ldots, n$$

Whereas again:

- β_0 is the constant (i.e., the predicted value if $x_{i1}, \ldots, x_{ip} = 0$)
- $\beta_1, \ldots, \beta_p$ are the regression coefficients (i.e., the effect of a one unit change in $X_1, X_2, \ldots, X_p$, ceteris paribus)
- ϵ_i is the individual error term (i.e., the difference between the true value y_i and the predicted value $\hat{y}_i$)

What is the best regression model to fit the data?

But… what means best?

- Ordinary least squares do not work here anymore
- Rather, the coefficients are estimated with the *Maximum Likelihood Approach* (ML)
- *Maximum Likelihood*:
  - Following an iterative process, the coefficients $\beta_0, \beta_1, \ldots, \beta_p$ are determined in such way that the observed values of Y are the most likely ones

Interpretation of the coefficients from a logistic regression I

Log Odd

- *Odds* versus *Probability*: using the example of a dice bet
  - Calculating the *probability* of having a 5 or a 6: $P(\{5,6\}) = 0.3\overline{3} = 33.\overline{3}\%$
  - Calculating the *odds* of having a 5 or a 6: $\frac{P(\{5,6\})}{P(\{1,2,3,4\})} = \frac{0.3\overline{3}}{0.6\overline{6}} = \frac{1}{2}$
  - Colloquial interpretation of odds: it is twice as likely to have a 1, 2, 3, or 4 than having a 5 or 6
- Back to the logistic regression:

$$\ln\left(\frac{P(y=1)}{1-P(y=1)}\right) = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_p * x_p + \epsilon_i$$

- How to interpret a change in X_j :
  - ➢ **a one-unit change in X_j will change the *Log Odd* by β_p**

Interpretation of the coefficients from a logistic regression II

Odds / Odds ratio

- Further transformation:

$$\ln\left(\frac{P(y=1)}{1-P(y=1)}\right) = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_p * x_p \xrightarrow{e^x} \frac{P(y=1)}{1-P(y=1)} = e^{\beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_p * x_p}$$

$$= e^{\beta_0} * e^{\beta_1 \cdot x_1} * e^{\beta_2 \cdot x_2} * \cdots * e^{\beta_p \cdot x_p}$$

- How to interpret a change in X_j:

$$\frac{P(y=1)}{1-P(y=1)}_{[x_p+1]} = e^{\beta_0} * e^{\beta_1 \cdot x_1} * e^{\beta_2 \cdot x_2} * \cdots * e^{\beta_p \cdot (x_p+1)} = e^{\beta_0} * e^{\beta_1 \cdot x_1} * e^{\beta_2 \cdot x_2} * \cdots * e^{\beta_p \cdot x_p} * e^{\beta_p} = \frac{P(y=1)}{1-P(y=1)}_{[x_p]} * e^{\beta_p}$$

$$\boxed{\frac{\frac{P(y=1)}{1-P(y=1)}_{[x_p+1]}}{\frac{P(y=1)}{1-P(y=1)}_{[x_p]}} = e^{\beta_p}}$$

➢ **a one-unit change in X_j will change the *Odds ratio* by the factor β_p**

Interpretation of the coefficients from a logistic regression III

Example: Dependent Var: voted for AFD in 2017 election | Independent Var: sex (1 – male, 0 – female)

Results

```
                           Logit Regression Results
==============================================================================
Dep. Variable:          secondvote_afd   No. Observations:                 2112
Model:                           Logit   Df Residuals:                     2110
Method:                            MLE   Df Model:                            1
Date:                 Wed, 20 Oct 2021   Pseudo R-squ.:                 0.02155
Time:                         19:00:22   Log-Likelihood:                -559.29
converged:                        True   LL-Null:                       -571.60
Covariance Type:             nonrobust   LLR p-value:                 6.933e-07
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     -3.0030      0.148    -20.307      0.000      -3.293      -2.713
male           0.8486      0.178      4.769      0.000       0.500       1.197
==============================================================================
```

→

```
In [192]: print(np.exp(results.params))
Intercept    0.049638
male         2.336343
dtype: float64
```

Odds ratio → 2.34:
Being male increases the odds of voting AFD by a factor of 2.34

Note: values above 1 imply a positive effect and values below 1 imply a negative effect

Log odd → 0.85:
Being male increases the log odd of voting AFD by 0.85

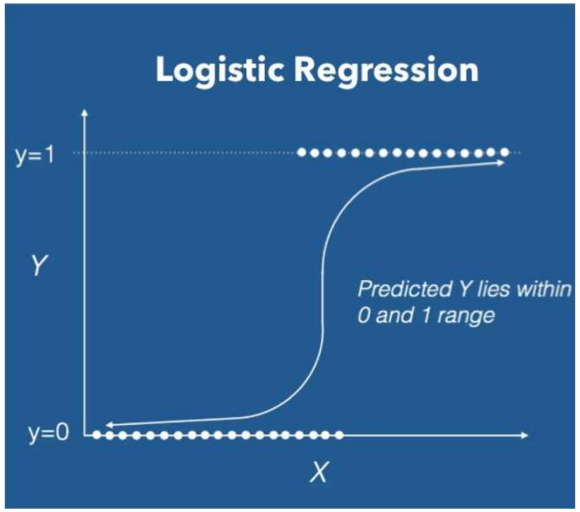
Note: As for linear regression, positive values imply a positive effect and negative values imply a negative effect

Problem: none of these values can be interpreted in terms of a difference in probabilities between male and female voters
Or in other words: these values remain relatively unaffected by the base rate of AFD voting

Interpretation of the coefficients from a logistic regression IV

Marginal effects

- Marginal effects allow statements in terms of probability differences
- Crucial point:
  - In contrast to log odds they are non-linear
  - They depend on the value of X_j and the other covariates $X_1, X_2, X_3, \ldots$
  - Different subjects will have different marginal effects
  - The reason lies in the fact that the probability estimation is bound between 0 an 1



- There are different form of marginal effects:
  1) Average Marginal Effect (AME): *The average of the marginal effects at each observation*
  2) Marginal Effect at the Mean (MEM): *The marginal effects at the mean of each regressor*
  3) Marginal Effects at Representative values (MER): *The marginal effects at specific values for the regressors*
- Marginal effects are calculated from the actual predictions that the model makes by using numerical methods (they are not analytically calculated!)

3. Logistic regression in Python

Libraries

- There are various ways to run a logistic regression in Python
- As for linear regression, we will again use **statsmodel**
- For instance, one such library is *statsmodel*:
  - Linear models
  - **Non-linear models**
  - Time series analysis
  - Event history analysis
  - Prediction and diagnostic
- Major advantage: enables the formula-style of R (`income ~ education + age + gender`)
- Extensive documentation for the library: https://www.statsmodels.org/stable/index.html

Even in statsmodel there are various modules and ways to run a regression. One way is by importing the following module:

```
import statsmodels.formula.api as smf
```

statsmodels

Running a logistic regression in *statsmodel*

1. Use the model class to describe the model
2. Fit the model using a class method
3. Inspect the results using a summary method

```
In [367]: model = smf.logit("secondvote_afd ~ male", data = vote_sub) # Describing the model

In [368]: results = model.fit() # Fitting the model
Optimization terminated successfully.
         Current function value: 0.264814
         Iterations 7

In [369]: print(results.summary()) # Summarizing the model
```

```
                           Logit Regression Results
==============================================================================
Dep. Variable:         secondvote_afd   No. Observations:                 2112
Model:                          Logit   Df Residuals:                     2110
Method:                           MLE   Df Model:                            1
Date:                Thu, 21 Oct 2021   Pseudo R-squ.:                 0.02155
Time:                        00:45:35   Log-Likelihood:                -559.29
converged:                       True   LL-Null:                       -571.60
Covariance Type:            nonrobust   LLR p-value:                 6.933e-07
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     -3.0030      0.148    -20.307      0.000      -3.293      -2.713
male           0.8486      0.178      4.769      0.000       0.500       1.197
==============================================================================
```

Predictions and confidence intervals

- With `results = model.fit()` we again create an object that contains several attributes such as the estimated parameters, the r square, but also the predicted values and the error terms (hint: type `dir(results)` to see a full list of all attributes)

- We can calculate marginal effects with `results.get_margeff()` and print the results by chaining additionally `….summary():”`
  - `get_margeff(at=”overall”)`: Average Marginal Effects (AME)
  - `get_margeff(at=”mean”)`: Marginal Effect at the Mean (MEM)

- We can further estimate the predicted probabilities for our observations with:

```
In [391]: predict = results.predict() # Accessing the predicted probabilities
```

- We can evaluate the quality of our model by calling the prediction table:

```
In [406]: predict_tab = results.pred_table(threshold=0.5) # threshold defines the classification probability
```

4. Hands on

… **Open *Session_4_logistic_regression.py***