Classification with Logit

Suppose we would like to investigate dependent variables that are binary, e.g. does the consumer buy the product or not, or will a start-up survive or not. For such cases, we have a binary dependent variable where we usually have a value of 1,if the event will occur and zero otherwise. We want to predict the probability of ouroutcome variable being equal to one or zero.

Why does OLS not work here?

- OLS requires a certain variation in the dependent variable, i.e. metric scale, which is not fulfilled for the binary outcome in our case.
- OLS assumes that the error term is normally distributed: This can be severely violated in our case.
- In addition, OLS will deliver predicted values that diverge from the 0 and 1 scheme we desire.

Logistic regression

In contrast to OLS,

- logistic regression does not deliver predicted values but probabilities of occurrence (of a 1).
- For this purpose, logistic regression creates a latent variable (Z). Z is not equal to our outcome variable but created artificially. This latent variable Z will then be determined by a linear function which looks like the OLS regression that we already know.
- The predicted values of our dependent variable (0/1) will be determined by Z. If the predicted value of Z is larger than zero, the predicted value of Y will be 1 and zero otherwise.

Example

Let us consider the “foodexport.xlsx” data

```
library(readxl)
fexp   <-   read_excel("D:/data/Empirical   Research/foodexport.xlsx")
head(fexp)
```

```
##         #         A         tibble:         6         ×         96
##   id      year CompanynameLatinalph…¹ BvDIDnumber MaterialcoststhEUR
RevenuethEUR
##   <chr> <dbl> <chr>                  <chr>                     <dbl>
<dbl>
## 1 FR00…  2016 Sarl Rigault et CIE    FR005650031               1140.
```

```
1758.
## 2 FR00…  2013 Forgez Pere et Fils    FR005720685              1243.
1632.
## 3 FR00…  2014 Forgez Pere et Fils    FR005720685              1176.
1661.
## 4 FR00…  2011 Laboratoire Nutergia   FR006380042              3638.
22803.
## 5 FR00…  2013 Laboratoire Nutergia   FR006380042              5257.
27314.
## 6 FR00…  2014 Laboratoire Nutergia   FR006380042              5451.
31541.
##     #     i     abbreviated     name:     ¹CompanynameLatinalphabet
##    #    i    90    more    variables:    CostsofemployeesthEUR    <dbl>,
Numberofemployees                                                   <dbl>,
##    #          ExportrevenuethEUR    <dbl>,    TotalassetsthEUR    <dbl>,
##    #        Dateofincorporation    <chr>,    NACERev2mainsection    <chr>,
##    #           NACERev2corecode4digits    <chr>,    naceprim    <dbl>,
##  #      NACERev2secondarycodes  <chr>,  Standardisedlegalform  <chr>,
## #   Nationallegalform <chr>, FixedassetsthEUR <dbl>, EBIT <dbl>, …
```

Let us check what determines the decision to export (ExportD=1) or not (ExportD=0)

```
logit                                                                =
glm(ExportD~Numberofemployees+logOmega+year_2012+year_2013+year_2014
           +year_2015+year_2016+year_2017+year_2018,
family="binomial", dat=fexp)

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(logit)

##
##                                                                Call:
## glm(formula = ExportD ~ Numberofemployees + logOmega + year_2012 +
##       year_2013 + year_2014 + year_2015 + year_2016 + year_2017 +
##              year_2018,   family   =   "binomial",   data   =   fexp)
##
##                                                          Coefficients:
##                              Estimate  Std.  Error  z  value  Pr(>|z|)
##  (Intercept)           -9.465e+00    1.525e-01  -62.079    <  2e-16  ***
##  Numberofemployees  -3.214e-04    8.679e-05    -3.703  0.000213  ***
##  logOmega                4.837e+00   6.666e-02   72.566   <  2e-16  ***
##  year_2012                 -1.081e-01    9.760e-02    -1.107  0.268146
##  year_2013               -2.261e-01   9.707e-02   -2.329  0.019860  *
##  year_2014                 -1.101e-01    9.490e-02    -1.160  0.246039
##  year_2015                 -1.231e-01    9.620e-02    -1.280  0.200563
##  year_2016                 -1.155e-01    9.757e-02    -1.184  0.236570
##  year_2017                 -9.050e-02    9.851e-02    -0.919  0.358298
##  year_2018                 -1.767e-02    1.002e-01    -0.176  0.860016
##                                                                   ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 32712  on 28738  degrees of freedom
## Residual deviance: 22617  on 28729  degrees of freedom
##   (10624 observations deleted due to missingness)
## AIC: 22637
##
## Number of Fisher Scoring iterations: 5
```

The resulting coefficients are not the impact on the probabilities but on the latent variable Z!However, (at least) we can interpret the sign of the coefficients and their significance.

To obtain the predicted probabilities, we can use the following piece of code

```
fexp$predictp = predict(logit, fexp, type = "response")
```

Let us see how accurate our prediction actually is.

We can say that we would predict that a firm exports if the predicted probability were >0.5

```
fexp$predictionexp = 0
for (i in 1:nrow(fexp)){
  if (is.na(fexp$predictp[i])){
    fexp$predictionexp[i]=NA }
  else if (fexp$predictp[i]>0.5) {
    fexp$predictionexp[i] = 1
  }

}
```

Now let us see what observations have been identified correctly

```
fexp$correct = 0
for (i in 1:nrow(fexp)){
  if (is.na(fexp$predictp[i])){
    fexp$correct[i]=NA }
  else if (fexp$predictionexp[i]==fexp$ExportD[i]) {
    fexp$correct[i] = 1
  }

}

summary(fexp$correct)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   1.000   1.000   0.825   1.000   1.000   10624
```

We predicted correctly the 82.5% of the cases.