Parametric Statistics

Week 13 - Logistic Regression

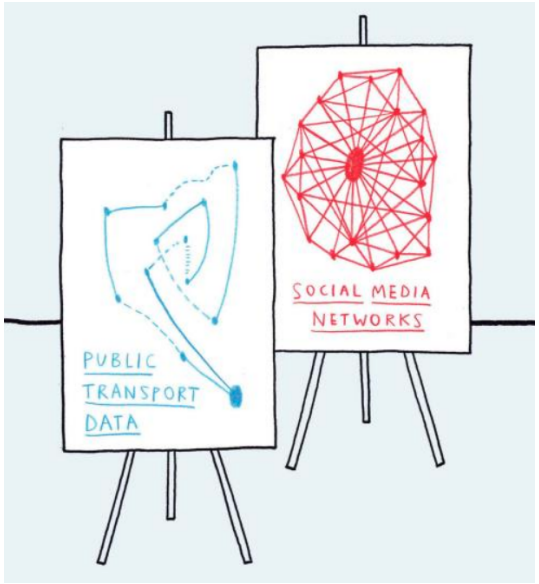
Juan Carlos Medina Serrano

Technische Universität München
Hochschule für Politik
Political Data Science

Munich 2020

political data science

https://politicaldatascience.blogspot.de



Categorial Outputs

Logistic regression is (like the name says) a regression. However the difference to linear regression is that the outcome variable is **categorical** not continuous.

The predictors can be continuous or categorical, similar to linear regression.

We will start by considering binary categorical variables (two-categories). Examples:

- Predict if a tumor is benign or malignant.
- Predict if a student will pass or fail an exam.
- Predict if in a picture there is a dog or a cat
- Predict if a video should appear on your YouTube main page or not
- Predict if an email is spam or not.

For binary variables, we give Y the values 0 and 1 (mathematical convention)

In machine learning, logistic regression is one of the many methods for **classification**. Most of the big IT companies use classification algorithms for their products and services.

Why not use linear regression?

Remember the (multiple) linear regression model:

$$\hat{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p$$

Why can't we use this model? Two reasons:

First: A main assumption for linear regression is linearity. This means a linear relationship between predictors (X1, X2...,Xp) with the outcome (Y). If the outcome is categorical there is no linear relationship anymore.

Second: We need a model that gives us **PROBABILITIES** to each class.
Example: We want to predict if a patient is sick or not. We measure a patients heart beat (X1) and blood pressure (X2), the model has to decide between Y=sick or Y=not sick. The model says that there is a 0.7 probability that the patient is sick and 0.3 that the patient is not sick. The doctor decides that the patient is sick because its the category with the **highest** probability.

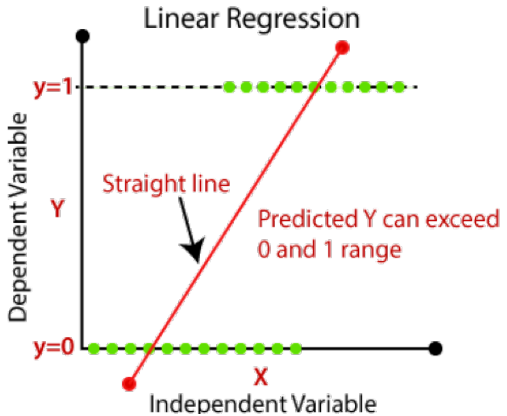
And why can't a linear regression model give probabilities?

Why not use linear regression?

And why can't a linear regression model give probabilities?
Because probabilities need to be between 0 a 1. A model like this:

$$P(Y) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p$$

where P is the probability of category Y. Will be a line that can output values that exceed 0 and 1.



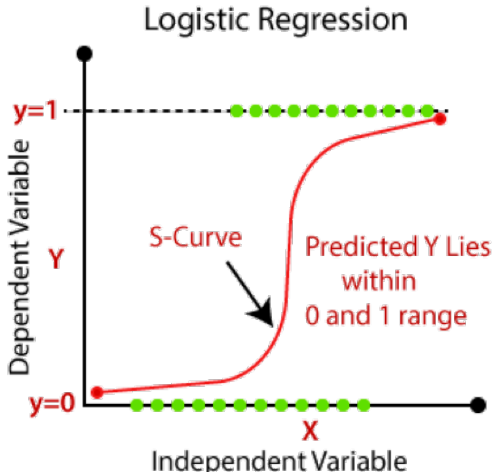
The green dots are individual cases, for example patients with a blood pressure (X) and are either healthy (Y=0) or sick (Y=1)

Logistic function

We need a model that can only give values between 0 and 1. This is the logistic function:

$$P(Y) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p)}}$$

Ohh.. this looks ugly...but hey the linear model is still there, only that now there is an **e** which is the **exponential** function (Google it if you forgot about it from high school). Now the model only outputs probabilities between 0 and 1:



Logistic function

$$P(Y) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p)}}$$

Don't be scared about the equation! The same rules from multiple linear regression apply here:

- You have coefficients βs for each predictor
- You can choose between different predictors (X1, X2, ...)
- You can build interactions (X1 * X2)
- You can include polynomial predictors

Linear regression and logistic regression are part of a family of methods called **Generalized Linear Models (GLMs)**. They all have these same rules. Only the function that surrounds the linear model changes.

Finding the Best Coefficients

We need a way to find the best coefficients. For linear regression we minimized the RSS. What should we minimize this time?

We have observed data (Y=0 or Y=1) and our model will predict if it belongs to each class. We need to sum the classification errors (all observations that are falsely classified). We use the **log-likelihood**:

$$loglikelihood = \sum [Y_i * log(P(Y_i)) + (1 - Y_i) * log((1 - P(Y_i)))]$$

Oh...another ugly equation, it's ok if you don't want to understand what it represents. But lets try it:

- In case Y=1 and we predict P(Y)=1. The first term is $Y * log(P(Y)) = 1 * log(1) = 1 * 0 = 0$ and the second term is $(1 - Y) * log((1 - P(Y))) = (1 - 1) * log(1 - 1) = 0$. Since our model predicted correctly there is no log-likelihood.
- In case Y=0 and we predict P(Y)=0. The first term is $Y * log(P(Y)) = 0 * log(0) = 0$ and the second term is $(1 - Y) * log((1 - P(Y))) = (1 - 0) * log(1 - 0) = 1 * log(1) = 1 * 0 = 0$. Since our model predicted correctly there is no log-likelihood.
- For wrong predictions the log-likelihood is not zero. The worse our prediction (Y=1 and we predict P(Y)=0.2) the bigger contribution to the log-likelihood. However, every contribution is **negative**! Since the logarithm function is always negative between 0 and 1.

Finding the Best Coefficients

To find the best coefficients we need to maximize the log-likelihood (with the RSS we wanted to minimize it). Taking the derivative and equal to zero.

HOWEVER, this time there is NO equation solution to the derivative. You read it right, derivation maths don't work with the log-likelihood so we need to find an approximate derivative.

We have computers that can do this! (All deep learning/neural networks use this process as well...calculating approximate derivatives...that is the big secret of AI and why we need powerful computers)

How good is the model?

If we have different models (different predictors), how do we select the best one?

Idea: Choose the model with highest log-likelihood!

Yes...but once again you need to take into account the number of predictors in your model...Better use AIC or BIC as before!

$$AIC = -2 * loglikelihood + 2P$$

$$BIC = -2 * loglikelihood + 2P * log(N)$$

where P are the number of predictors.

The model with the lowest AIC/BIC ist the best one for our data.

The term $-2 * loglikelihood$ has a name, its called the **deviance**.

Model Significance

In multiple linear regression, we applied the F-test for model significance and individual t-tests for each predictor.

For logistic regression, we will use the χ^2 test for the model and z-tests for the predictors. The z-statistic is similar as the t-statistic, but uses the normal distribution for calculating critical and p values.

$$z = \frac{\beta}{SE}$$

where SE is the standard error of a predictor.

R Code

The code for logistic regression is similar to the linear regression. We use now **glm** instead of **lm** and we add a parameter *family=binomial*

$$glm(Y \sim X1 + X2, data = yourdata, family = binomial)$$

R Code

Example: Model predicts if a patient is cured (Y) after an intervention (X1) and the duration of a treatment (X2):

```
glm(formula = Cured ~ Intervention + Duration, family = binomial(),
    data = eelData)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6025  -1.0572   0.8107   0.8161   1.3095

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)         -0.234660   1.220563  -0.192  0.84754
Intervention         1.233532   0.414565   2.975  0.00293 **
Duration            -0.007835   0.175913  -0.045  0.96447
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 154.08  on 112  degrees of freedom
Residual deviance: 144.16  on 110  degrees of freedom
AIC: 150.16
```

The intervention is significant, but the duration of the treatment not. The null deviance corresponds to the deviance without any predictor and the residual deviance when the predictors are included. The χ^2 statistic can be calculated from these two values. (R does not show the value, you need to calculate it on your own, check the Literature to find out how)

Making Sense of the β coefficients

In linear regression, a β coefficient had this meaning: An increase of one unit of a predictor X, will increase the outcome Y by β.

For logistic regression, what does an increase of predictor X means? With some math we convert the logistic function into:

$$log(\frac{P(Y)}{1-P(Y)}) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p$$

The quantity on the left side is the logarithm of the **odds**. The odds of an event is the probability that the event happens divided by the probability that it does not happen.

Increasing one unit of X, the logarithm of odds will increase by β. This is the same as saying that increasing one unit of X, the odds will increase by e^{β}.
If e^{β} is bigger than one, the predictor increases the odds of the event. On the other hand, if its smaller than one, the predictor decreases the odds of the event.

For the example in the last slide, the coefficient for Intervention is 1.23, and the exponential is $e^{1.23} = 3.42$. So the odds of being cured increases by 3.4 with an increase of one unit of Intervention.

Multicollinearity

Multicollinearity is still a problem for logistic regression! Check for it for every model you make.

Multinomial Logistic Regression

We have explored logistic regression with only two categories. What if we want to predict between more than two categories?

Multinomial logistic regression allows to predict more than two categories and it uses the same principles we learned so far. No need to introduce new equations!!!

It works by breaking the outcome variable down into a series of comparisons between two categories. For example, for three categories A,B and C, we need to do two comparisons. A logistic regression model between A and B, and a logistic regresion between A and C. For this, you need to choose the base category A!

In R,
the function you need is the **mlogit**, which works almost the same as glm, only that you need to add the base category.

Literature

- Discovering Statistics using R: Chapter 8