TMA4315 Generalized Linear Models

Assignment 2:

GLMs for count data and binomial data

Deadline: November 5, 2014

Send your report and code electronically to `fuglstad@math.ntnu.no`.

Guidance will be by appointment and there will be no lectures in week 43 and Tuesday in week 44. The next lecture is Thursday October 30. If you need help with the exercise, contact Geir-Arne Fuglstad at `fuglstad@math.ntnu.no` or Håvard Rue at `hrue@math.ntnu.no`.

In Assigment 1 you made your own `myglm`-package to handle Gaussian responses. In this exercise you will extend this code to also handle binomial responses and Poisson responses within the same framework. You will see that the differences are that it is now necessary to perform numerical optimization to find the parameter estimates and covariance estimates, and that residual sum of squares no longer is a useful concept and that we must work with deviances instead.

Exercise 1: Poisson regression

The dataset given in `smoking.txt` consists of four variables:

age: in five-year age groups 40-44, 45-49, 50-54, 55-59, 60-64, 65-69, 70-74, 75-79, 80+.

smoking status: doesn't smoke, smokes cigars or pipe only, smokes cigarrettes and cigar or pipe, and smokes cigarrettes only.

population: in hundreds of thousands.

deaths: number of lung cancer deaths in a year.

We are interested in studying if the mortality rate due to lung cancer (the number of deaths due to lung cancer per $100,000$ individuals during one year) controlled for age group varies with smoking status. Assume that the number of deaths for each set of covariate values, Y_i, can be considered Poisson distributed, $Y_i \sim \text{Poisson}(\mu_i)$.

a) One of the variables are different than the others and should be treated as an offset. Which one and why? How should it enter in μ_i?

b) We model the log-mortality, ν_i, in $\log(\mu_i) = \text{offset}_i + \nu_i$ with a linear model and end up with the standard Poisson GLM. Write up the likelihood as a function of the parameters β.

c) Extend the `myglm` package from Assigment 1 so that it can fit this model.

   Suggestions:

   1. Add an additional argument `family` to your `myglm` function and make it use the code from Assigment 1 if `family = "gaussian"` and include new code for the case `family="poisson"`.

2. If the formula is written `y~offset(log(x1))+x2`, the offset can be extracted with `offset = model.offset(mf)` from the model frame object.
3. Use the `R`-function `optim` to find the maximum likelihood estimates for β.
4. Use `hessian = TRUE` in `optim` so that it also returns the Hessian at the mode. Calculate the estimated covariance matrix based on this Hessian.
5. Note: You might have to implement a function that evaluates exact derivatives of the likelihood and provide it to `optim` if you want to get the same results as the standard `glm` function

d) Fit the full model using additive effects. Is the model satisfactory? Is smoking a significant factor? (Consider the deviances of the models)

Exercise 2: Logistic regression

In credit scoring the goal is to determine if a customer should be given credit or not. If credit is given to someone who can not service it, the company loses money, and if credit is not given to someone who would have been able to service it, less money is earned. The file `credit_scoring.txt` consists of 1000 entries each consisting of 20 explanatory variables and an indicator for whether the customer is good or bad. The goal of this exercise is to make a model for predicting whether a customer should be given credit (is good) or should not be given credit (is bad). The file `credit_scoring_description.txt` contains descriptions of the different variables and the values they can take.

In this case the goal is to predict as well as possible and how good a model is is determined by how well it predicts. You should consider classifying a

bad customer as good five times as expensive as classifying a good customer as bad. The predictive power should be evaluated based on 10-fold cross-validation. This is not covered in the lecture, and you have to find material on cross-validation and classification errors online.

The main tool to use is a binomial model where you model logit-transformed probabilities with a linear model,

$$y_i|p_i \sim \binom{n_i}{y_i} p_i^{y_i}(1-p_i)^{n_i-y_i}$$

and

$$\text{logit}(p_i) = \mathbf{X}\beta,$$

where $\mathbf{X}$ is the model matrix and β is the model parameters. Make changes to your `myglm` package as they are required to solve this problem.

A full solution to this exercise should contain

- Code for fitting the binomial GLM
- Estimation of the classification error by cross-validation
- Description of how you selected the final model (e.g. comparisons of deviances and classification errors)
- A description of the final model and the intepretation of the parameters
- A discussion of whether your final model is working well or not