MSc Business Analytics: **Quantitative Methods**

Exercises

The following exercise offers you the opportunity to test your understanding of the materials presented to you in Lectures 8 and 9.

Residual checking for non-Gaussian error models is not always as straightforward as it is in the Gaussian case, and the problems are particularly acute in the case of binary data. This question explores this issue.

- The following code fits a GLM to data simulated from a simple binomial model and examines the default residual plots.

  ```
  n <- 100; m <- 10
  x <- runif(n)
  lp <- 3*x-1
  mu <- binomial()$linkinv(lp)
  y <- rbinom(1:n,m,mu)
  par(mfrow=c(2,2))
  plot(glm(y/m ~ x,family=binomial,weights=rep(m,n)))
  ```

  Run the code several times to get a feel for the range of results that are possible even when the model is correct (as it clearly is in this case).

- Explore how the plots change as m (the number of binomial trials) is reduced to 1. Also examine the effect of sample size n.

- By repeatedly simulating data from a fitted model, and then refitting to the simulated data, you can build up a picture of how the residuals should behave when the distributional assumption is correct, and the data are really independent. Write code to take a `glm` object fitted to binary data, simulate binary data given the model fitted values, refit the model to the resulting data, and extract residuals from the resulting fits. Functions `fitted` and `rbinom` are useful here.

- If `rsd` contains your residual vector then

  ```
  plot(sort(rsd),(1:length(rsd)-.5)/length(rsd))
  ```

  produces a plot of the 'empirical CDF' of the residuals, which can be useful for characterizing their distribution. By repeatedly simulating residuals, as in the previous part, you can produce a 'simulation envelope' showing, e.g., where the middle 95%of these 'empirical CDFs' should lie, if the model assumptions are met: such envelopes provide a guide to whether an observed 'real' residual distribution is reasonable, or not. Based on your answer to the previous part, write code to do this.