Modeling in Revolution R Enterprise
Module 4: Generalized Linear Models

Generalized Linear Models

Let’s continue our discussion of regression and model construction, only this time considering a different type of data with a non-normal error distribution. Perhaps before we start this, let’s talk a little about the distribution of our response variables first.

Let’s consider a few examples for some response variables that we might model:

- Response times to reactions from a medicine
- Measurements of heights for a certain bird species
- Carbon present in an object of historical interest

While each of these examples may appear no different from the other, important factors are present that we must take into consideration so as to maximize the effectiveness of our statistical

Normal Distribution

Let's again consider the measurement example. Let's say we are trying to measure the heights of certain birds in a conservatory, and we are using a ruler to collect such data. We will find that:

- A few birds will be taller then usual
- A few birds will be shorter then usual
- The vast majority of birds will be somewhere between these extremes

In the end, we may assume that our response variable follows a normal distribution like the graph on the following page:

```
x <- seq(-3, 3, length = 100)
y <- dnorm(x)
plot(x, y, type = "l", col = "blue")
```

Normal Distribution

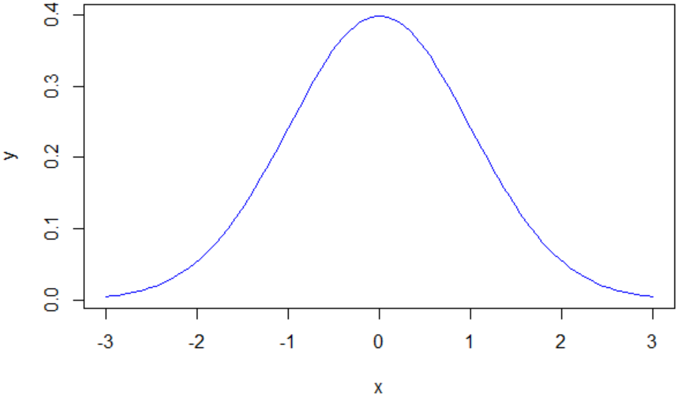


Figure:

Non-Normal Response Variables

However, not all response variables are normally distributed. One popular branch of statistics is survival analysis, which considers the amount of time until an expected event (or events) occur.

For instance, let’s reconsider our response time to a certain medicine example above, but extend it a little. Let’s say that we are trying to model the recovery rate from a disease after the appropriate medicine has been administered. In this case, patients would drop out of the study after fully recovering.

Non-Normal Response Variables

In this case, several non-normal distributions may accurately describe our data. For our example, let's suppose that the response variable follows the Exponential distribution. Then, our variables would look something like the following:

```r
x <- seq(0, 5, length = 50)
plot(x, dexp(x, 1), col = "red", type = "l")
```

Notice that we can think of the x-axis showing the time elapsed, and the y axis shows the proportion of patients in our study.

Exponential Distribution

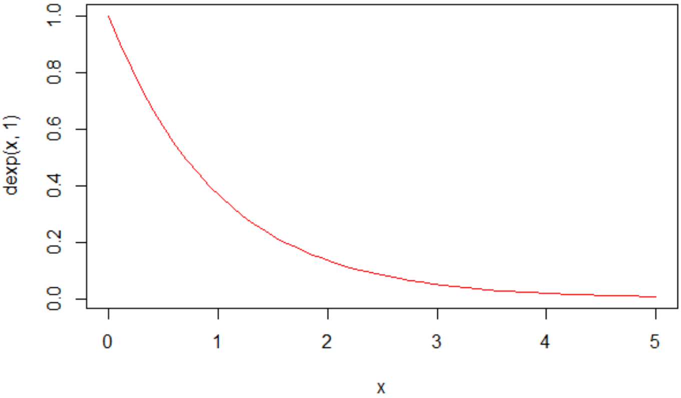


Figure:

Additional Forms for Big Data GLMs

General Linearized Models (GLMs) are an extension of ordinary linear regression but tolerate response variables with error distribution models different from the normal distribution.

Families for distribution that are executed in C++ are: binomial/logit, gamma/log, poisson/log, and Tweedie. Other family/link combinations use a combination of C++ and R code.

rxGlm

The rxGlm command fits a generalized linear model to a set of data by specifying a model formula, a root data set, and a family.

```
rxGlm(formula, data, family = gaussian(), ...)
```

Notice that family specifies the type of distribution that characterizes the response variable.

Example: Big Data GLMs

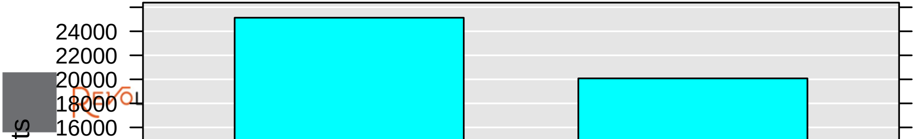
This time we will be modeling a response variable that is characterized by the Bernoulli distribution, indicating the presence of a yes or no response from a client. In this case, a linear model would be less suitable than a Generalized Linear Model, due to the restrictive boundaries of the response variables.

Example: Big Data GLMs

Considering our Bank data, notice that housing is characterized as a Bernoulli variable.

```r
infile <- file.path("data", "BankXDF.xdf")
BankDS <- RxXdfData(file = infile)
rxHistogram(~housing, data = BankDS)
```

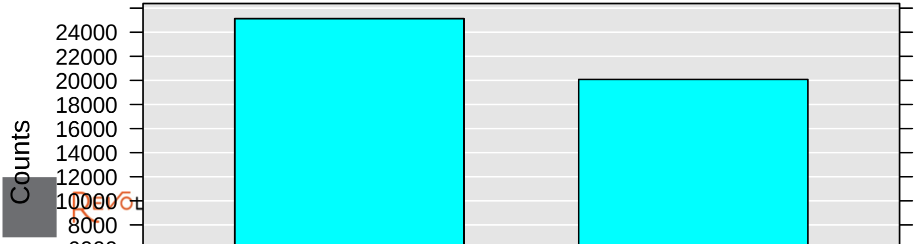
```
## Rows Read: 10000, Total Rows Processed: 10000, Total Chunk Time: 0.001 seconds
## Rows Read: 10000, Total Rows Processed: 20000, Total Chunk Time: 0.001 seconds
## Rows Read: 10000, Total Rows Processed: 30000, Total Chunk Time: Less than .001 seconds
## Rows Read: 10000, Total Rows Processed: 40000, Total Chunk Time: Less than .001 seconds
## Rows Read: 5211, Total Rows Processed: 45211, Total Chunk Time: Less than .001 seconds
## Computation time: 0.008 seconds.
```



Example: Big Data GLMs

```
infile <- file.path("data", "BankXDF.xdf")
BankDS <- RxXdfData(file = infile)
rxHistogram(~housing , data = BankDS)
```

```
## Rows Read: 10000, Total Rows Processed: 10000, Total Chunk Time: 0.001 seconds
## Rows Read: 10000, Total Rows Processed: 20000, Total Chunk Time: 0.001 seconds
## Rows Read: 10000, Total Rows Processed: 30000, Total Chunk Time: Less than .001 seconds
## Rows Read: 10000, Total Rows Processed: 40000, Total Chunk Time: Less than .001 seconds
## Rows Read: 5211, Total Rows Processed: 45211, Total Chunk Time: Less than .001 seconds
## Computation time: 0.009 seconds.
```



Example: Big Data GLMs

Let’s model the case of whether a client owns a house, based on the response variables balance and marital status. Note that in this case, we need the Binomial distribution which generalizes to the Bernoulli distribution when there is only one parameter:

```
glmMod <- rxGlm(housing ~ balance + marital, data = BankDS, family = binomial())

## Rows Read: 10000, Total Rows Processed: 10000, Total Chunk Time: 0.002 seconds
## Rows Read: 10000, Total Rows Processed: 20000, Total Chunk Time: 0.007 seconds
## Rows Read: 10000, Total Rows Processed: 30000, Total Chunk Time: 0.006 seconds
## Rows Read: 10000, Total Rows Processed: 40000, Total Chunk Time: 0.005 seconds
## Rows Read: 5211, Total Rows Processed: 45211, Total Chunk Time: 0.004 seconds
##
...
```

Note that the same model can also be fit using the rxLogit function, which is intended for the case of the binomial family with the “logit” link family:

Example: Big Data GLMs

Analyzing the summary output…

```
summary(glmMod)
```

```
## Call:
## rxLogit(formula = housing ~ balance + marital, data = BankDS)
##
## Logistic Regression Results for: housing ~ balance + marital
## Data: BankDS (RxXdfData Data Source)
## File name: data/BankXDF.xdf
...
```

Example: Big Data GLMs

Notice that all variables, with the exception of the single case in marital status, is significant. Let's interpret these variables below.

Example: Big Data GLMs

In this case, our interpretation will be similar to the outcome of the logistic regression model. We will use the logit link function, equating the results using the probability *p* as the success probability based on the response variables (i.e. the probability of owning a house, which is a binary response):

$$logit(p(balance, marital_{single}, marital_{married})) = \frac{log(p(balance, marital_{single}, ma}{1 + p(balance, marital_{single}, m}$$

Example: Big Data GLMs

If we define:

$$\begin{aligned} \mathit{Intercept} &= \beta_0 = -0.29343434 8 \\ \beta_{\mathit{balance}} &= \beta_1 = 0.000050759 \\ \beta_{\mathit{married}} &= \beta_2 = -0.031799445 \\ \beta_{\mathit{single}} &= \beta_3 = 0.069158524 \end{aligned}$$

Where the probability function is defined for our model as:

$$p(\mathit{balance}, \mathit{marital}_{\mathit{single}}, \mathit{marital}_{\mathit{married}}) = \frac{e^{\beta_0 - \beta_1 \mathit{balance} + \beta_2 \mathit{marital}_{\mathit{single}} + \beta_3 \mathit{marital}_{\mathit{marrie}}}}{1 + e^{\beta_0 - \beta_1 \mathit{balance} + \beta_2 \mathit{marital}_{\mathit{single}} + \beta_3 \mathit{marital}_{\mathit{m}}}}$$

Example: Big Data GLMs

Notice that the exponential is just a linear combination of the significant variables along with their beta coefficients.

The logit function is equal to the linear combination of the significant variables and their beta coefficients as well:

$$\begin{aligned} &\mathit{logit}(p(\mathit{balance}, \mathit{marital}_{\mathit{single}}, \mathit{marital}_{\mathit{married}})) \\ &= \beta_0 - \beta_1 \mathit{balance} + \beta_2 \mathit{marital}_{\mathit{single}} + \beta_3 \mathit{marital}_{\mathit{married}} \end{aligned}$$

Interpretations

Interpreting our model, it again makes more sense to consider a response variable increase from *x* to $(x+1)$, using a ratio of predicted responses. For example, for a one unit increase in balance (equivalent to an additional holding an additional euro):

$$\frac{e^{\beta_0+\beta_1(\mathit{balance}+1)+\beta_2(\mathit{marital}_{\mathit{divorced}})+\beta_3\mathit{marital}_{\mathit{married}}}}{e^{\beta_0+\beta_1(\mathit{balance})+\beta_2(\mathit{marital}_{\mathit{divorced}})+\beta_3\mathit{marital}_{\mathit{married}}}} = e^{\beta_1}$$

Similarly, the derivations for the other one unit changes in variables result in the same responses.

Interpretations

```
df <- as.data.frame(glmMod$coefficients)
df[2] <- exp(df[1])
names(df) <- c("coeff", "oddsRatio")
df
```

```
##                       coeff oddsRatio
## (Intercept)      -2.934e-01    0.7457
## balance           5.076e-05    1.0001
## marital=married  -3.180e-02    0.9687
## marital=single    6.916e-02    1.0716
## marital=divorced         NA        NA
```

Interpretations

Accordingly,

- For a one unit increase in balance, equivalent to a client holding an additional euro, the probability that the client owns a house increases by a factor of $e^{0.00005075911} = 1.0000508$.
- If the client is married, the probability that the client owns a home decreases by a factor of $e^{-0.03179944531} = 0.9687008$.
- If the client is single, the probability that the client owns a home increases by a factor of $e^{0.06915852449} = 1.0716061$.

Exercise: Generalized Linear Models

For this exercise, model the probability of a client having credit in default (default, hint: Bernoulli distribution) using response variables:

- balance,
- age,
- and housing.

Afterwards, give interpretations for each of the significant variables.

Exercise: Solution

Constructing the Generalized Linear Model with a Bernoulli family, then we can use the rxLogit function:

```
glmMod <- rxLogit(default ~ balance + age + housing, data = BankDS)
```

```
## Rows Read: 10000, Total Rows Processed: 10000, Total Chunk Time: 0.002 seconds
## Rows Read: 10000, Total Rows Processed: 20000, Total Chunk Time: 0.003 seconds
## Rows Read: 10000, Total Rows Processed: 30000, Total Chunk Time: 0.003 seconds
## Rows Read: 10000, Total Rows Processed: 40000, Total Chunk Time: 0.003 seconds
## Rows Read: 5211, Total Rows Processed: 45211, Total Chunk Time: 0.003 seconds
##
...
```

```
summary(glmMod)
```

```
## Call:
## rxLogit(formula = default ~ balance + age + housing, data = BankDS)
##
## Logistic Regression Results for: default ~ balance + age + housing
## Data: BankDS (RxXdfData Data Source)
## File name: data/BankXDF.xdf
...
```

Exercise: Solution

```
df <- as.data.frame(glmMod$coefficients)
df[2] <- exp(df[1])
names(df) <- c("coeff", "oddsRatio")
df
```

```
##                 coeff oddsRatio
## (Intercept) -2.686468   0.06812
## balance     -0.002357   0.99765
## age         -0.011885   0.98818
## housing=yes -0.457808   0.63267
## housing=no         NA        NA
```

Exercise: Solution

Interpreting the significant variables using the same process as was used in the previous example, we have:

- For a one unit increase in balance (equivalent to holding one additional euro), the probability that the client has credit in default decreases by a factor of $e^{-0.002357039} = 0.9976457$.
- For a one unit increase in age, the probability that the client has credit in default decreases by a factor of $e^{-0.011885470} = 0.9881849$.
- If the client does own a home, then the probability that the client has credit in default decreases by a factor of $e^{-0.457807517} = 0.6326692$.

Recap

Let’s review some of the concepts covered in this module:

- How do GLM’s differ from Linear or Logistic Regression?
  - In what ways can it be similar?
- What are some of the names of family distributions for GLM’s?
- How do you interpret logit output from a GLM?
  - How is this similar to Logistic Regression?

Thank you

Revolution Analytics is the leading commercial provider of software and support for the popular open source R statistics language.

www.revolutionanalytics.com, 1.855.GET.REVO, Twitter: @Revo-lutionR