Heart Disease Regression Analysis

Sean O'Malley

Regression Methodologies: Discussion and Overview

Contrast between logistic regression and linear regression

Linear Regression requires the dependent variable to be continuous numeric values, while logistic regression requires the dependent variable to be categorical. Binary logistic regression is self explanatory for having a binary categorical option, while multinomial or ordinary logistic regression can have dependent variable.

Linear regression is based on least square estimation, which says regression coefficients should be chosen in such a way that it minimizes the sum of squared distances of each observed response to its fitted value. On the other hand, logistic regression is based on maximum likelihood estimation, which says coefficients should be chosen in such a way that it maximizes the probability of Y given X. In machine learning, the computer uses different "iterations" in which it tries different solutions until it gets the maximum likelihood estimates.

Graphically: Linear regression aims at finding the best fitting straight line, which is also called a regression line. In the above figure, the red diagonal line is the best fitting straight line and consists of the predicted score on Y for each possible value of X. The distance between the points to the regression line represent the errors.

Graphically: Logistic regression graphical output usually follows an S curve. Changing the coefficient leads to change in both the direction and the steepness of the logistic function. It means positive slopes result in an S-shaped curve and negative slopes result in a Z-shaped curve.

Linear regression needs a linear relationship between the dependent variables, while logistic regression does not require relationship between dependent and independent variables.

Linear regression requires error term should be normally distributed, while logistic regression does not require error term should be normally distributed.

Linear regression assumes that residuals are approximately equal for all predicted dependent variable values, while logistic regression does not need residuals to be equal for each level of the predicted dependent variable values.

What are logistic regression, multinomial regression, and polynomial regression good for?

Linear regression is used to estimate the dependent variable incase of a change in the independent variable, whereas logistic regression is used to calculate the probability of an event.

Polynomial regression is the best fit line that is not straight like linear regression, it is rather a curve that fits into the data points. In polynomial regression we have to be careful with overfitting the data in regards to creating a solid distribution, but also in terms of using polynomial regression for predictions.

Multinomial regression is used to describe data and to explain the relationship between one dependent nominal variable and one or more continuous-level(interval or ratio scale) independent variables.

Stepwise regression is a form of regression used when we deal with multiple independent variables. The selection of independent variables is done with the help of an automatic process, which involves no human intervention. This is achieved by observing statistical values like R^2, t-stats, and AIC metric to discern significant variables. Stepwise regression basically fits the regression model by adding/dropping co-variates one at a time based on a specific criterion.

Ridge regression is a technique used when the data suffers from multicollinearity (dependent variables and independent variables are highly correlated). Ridge regression solves multicollinearity problem through the shrinkage parameter. By adding this to the equation of the least sum of squares, ridge regression can shrink the parameter to have a very low variance.

Lasso regression (least absolute shrinkage selection operator), similar to ridge regression, also penalizes the absolute size of the regression coefficients, and is also capable of reducing the variability and improving accuracy of linear regression models. Lasso regression differs from ridge regression in that it uses absolute values in the penalty function instead of squares. This leads to penalizing values which causes some of the parameter estimates to turn out exactly zero. Larger the penalty applied, further the estimates get shrunk towards absolute zero. Larger the penalty applied, further the estimates get shrunk towards absolute zero. This results to variable selection out of the given variables.

Can you give some real life examples for logistic regression, multinomial regression, and polynomial regression?

Logistic: A researcher is interested in how variables, such as GRE, GPA, and prestige of the undergraduate institution, effect admission into graduate school. The response variable, admit/don't admit, is a binary variable.

Multinomial: A biologist may be interested in food choices that alligators make. Adult alligators might have different preferences from young ones. The outcome variable here will be the types of food, and the predictor variables might be size of the alligators and other environmental variables.

Polynomial: How is the length of a bluegill fish related to its age? How does your birthweight relate to your adult height?

What is the R function for fitting the logistic model?

In logistic regression we split the cleaned data into two chunks: training and testing datasets. The training will be used to fit our model which one then tests over the testing dataset.

```
train <- data[1:800,]
```

```
test <- data[801:889,]
```

To fit the model, lets use a binomial logistic regression for example, we use the glm function.

```
model <- glm(indvars ~., group = binomial(link = 'logit'), data = train)
```

We then use the summary function to obtain the results of the model

```
summary(model)
```

How do you interpret the coefficients from the R output?

We first note the statistical significance of the coefficients used in the logistic regression analysis via the p values. Paying notice to the sign of the coefficients is also important, negative coefficients suggest an inverse effect of that coefficient on the dependent variable.

Many also use anova() to interpret output and efficiency of a model. This is useful in noting the difference in noting the variance in effectiveness of the model to both fit and not overfit the model at hand.

Some also use the pscl package for the pR2 function to better understand model efficiency. Though there is no direct direct equivalent to R^2 in linear regression. The pR2 function gives us McFadden's R^2 index to assess model fit, allowing us to better determine the efficiency of our model in understanding the variance at play.

Binary Logistic Regression: Heart Disease Detection

```
heart <- read.csv("/Users/SeanOMalley1/Desktop/MSDS_660_Stats/chd_heart_data.csv")
```

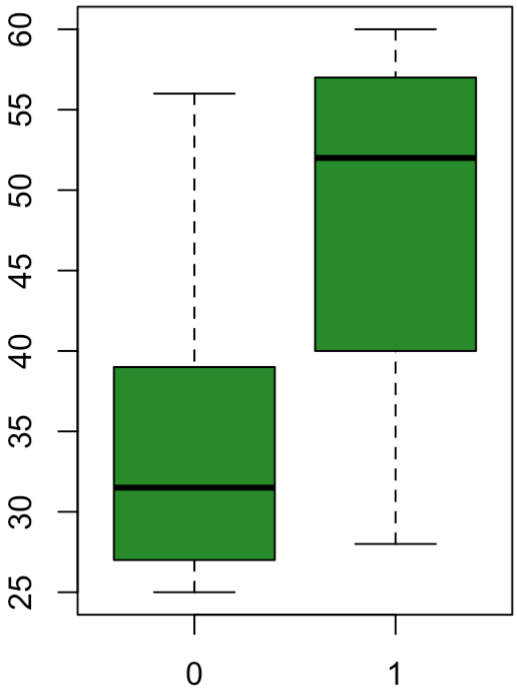
EDA

```
glimpse(heart)
```

```
## Observations: 23
## Variables: 3
## $ Patient <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,...
## $ Age     <int> 25, 26, 28, 30, 31, 32, 34, 35, 36, 27, 39, 40, 50, 51...
## $ CHD     <int> 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, ...
```

```
summary(heart)
```

```
##     Patient          Age             CHD        
##  Min.   : 1.0   Min.   :25.00   Min.   :0.0000  
##  1st Qu.: 6.5   1st Qu.:31.50   1st Qu.:0.0000  
##  Median :12.0   Median :40.00   Median :1.0000  
##  Mean   :12.0   Mean   :42.96   Mean   :0.5652  
##  3rd Qu.:17.5   3rd Qu.:54.50   3rd Qu.:1.0000  
##  Max.   :23.0   Max.   :60.00   Max.   :1.0000
```



Hypothesis Development

H^0 : Coronary heart disease is not significantly effected by age of the patient

H^1 : Age is a significant factor in the presence of coronary heart disease

Hypothesis Testing

I decided not to use the norm of the test and train sampling groups for this logistic regression, because with a dataset of 23, the sampling, even if random, would provide for trouble confirming the validity of the model. Therfore I will use other methods to best determine the validity of the binary logistic regression model.

Build the Model

```
model <- glm(formula = CHD ~ Age, family = binomial(link = "logit"), data = heart)
```

Test the Model Effectiveness

```
summary(model)
```

```
##
## Call:
## glm(formula = CHD ~ Age, family = binomial(link = "logit"), data = heart)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9524  -0.8104   0.4895   0.7037   1.7148
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.16311    1.88520  -2.208   0.0272 *
## Age          0.10550    0.04443   2.375   0.0176 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 31.492  on 22  degrees of freedom
## Residual deviance: 24.191  on 21  degrees of freedom
## AIC: 28.191
##
## Number of Fisher Scoring iterations: 4
```

Looking at the **glm()** output, we see that age is statistically significant in the presence of CHD in a patient, this was confirmed by the p-value. The positive coefficient of age suggests that all factors being equal, the higher the age of the person, the greater chance of them having CHD.

We can additionally gain knowledge using the **anova()** function to analyst the table of deviance.

```
anova(model, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: CHD
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                    22     31.492
## Age   1   7.3009        21     24.191 0.006892 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The difference between the null deviance and the residual shows how our model is doing against the null model. The wider the gap between these numbers, the better. Looking at the residual deviance from null to age we see that our model does make a significant improvement on our ability to predict the binary outcome of the presence of CHD.

We an also do our best to find the amount of variance explained via the **pscl** package and McFaddens R^2

```
pR2(model)
```

```
##         llh     llhNull          G2    McFadden        r2ML        r2CU
## -12.0957118 -15.7461744   7.3009252   0.2318317   0.2719835   0.3647368
```

Contrary to the anova and summary output, McFadden's R^2 appears to suggest that we are not explaining much of the variance of the model. I believe that this has much to do with the small sample size and lack of massive precision needed in binary logistic regression. Nevertheless, lets see how well we can predict with this model.

Assess Predictive Ability of the Model

Once again, due to small sample size I thought it would be best to feed the prediction model 3 new datatypes rather than subset of the previous data. I honestly disagree with using a sample size this small for too much hard validity, so I am willing to sacrifice some degree of testability for anything that would help improve a poorly fed model.

```
fitted.results <- predict(model, newdata = subset(heart, select = c(1,2,3)), type = "res
ponse")

fitted.results <- ifelse(fitted.results > 0.5, 1, 0)

misClasificError <- mean(fitted.results != heart$CHD)

print(paste('Accuracy', 1-misClasificError))
```

```
## [1] "Accuracy 0.782608695652174"
```

Now, understanding that we used a subset of the original data to test the original model, with the threat of overfitting and overlap, I do however believe the model is more accurate keeping the entire, very small, dataset. We see an accuracy of 78%, which given a binary outcome seems to work very well given what we had to work with.

In conclusion, I would suggest that we can reject the null hypothesis, that coronary heart disease is not significantly effected by age of the patient. This is confirmed by multiple model tests and finally by our prediction accuracy numbers.