Logistic Regression in R

In this lab, we are going to be working through an example of logistic regression to get a basic understanding of how we can do this in R. Before getting into the code, it is important to understand what type of situation we are modeling, and how we are going about modeling the situation.

The definition of **logistic regression** to most is the following:

- We want to model a binary outcome outcome by linking a log-odds ratio to a linear combination of predictor variables.
- We estimate our coefficients using maximum likelihood (which is different than what we do for multiple linear regression in 6530 and 6620).
- Maximum likelihood is achieved by using an algorithm known as Fisher's Scoring algorithm.

Note, that both linear regression as you have seen before (with a normally distributed response) and logistic regression, as described above, are two potential techniques that are a subset of what are known as **Generalized Linear Models**. You may also choose a different distribution for the response variable (Poisson and Multinomial distributions are very common), which would create another case of a generalized linear model. For the purpose of this document, we will focus on how the logistic model outlined above works. This is also the model discussed in the text.

The idea of logistic regression suggests we have a response of the following form:

$$y_i \sim Bernouilli(\pi_i)$$

and

$$log(\frac{\pi_i}{1-\pi_i}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots$$

Here, we assume that the y_i are independent of one another. The log odds used above is called the 'link' function, as it links the response to the explanatory variable(s). There are other link functions, but the log-odds is almost always used to map in logistic regression. Note y_i in this case

$$y_i = \begin{cases} 1, & \text{if success} \\ 0, & \text{otherwise} \end{cases}$$

When we interpret the results of logistic regression, we generally do so in terms of the odds. Similar to in linear regression the way we discussed in the previous lab, we talk about how a given x-variable influences the odds with regard to the coefficient attached to a particular x-variable. The output retrieved from logistic regression looks very similar to linear regression in terms of the estimates and p-values retrieved from the output.

A direct interpretation of an estimate of a particular β ($\hat{\beta}$) can be stated as: when x_i increases by 1 unit, the predicted change in the odds of success is a multiplicative change of $\exp(\hat{\beta}_i)$. If you are working with a categorical random variable, you are interpreting the coefficient as it compares to the baseline, similar to the way you would make an interpretation in linear regression.

To see these items in practice we will look at the RHTML document as a continuation of the material discussed above. Before you start into code, a few last notes:

- We often use logistic regression because we have nice interpretations of our coefficients. Still, adding higher order terms, takes away from our interpretability of coefficients.
- With large sample sizes, all of our predictors come back as *statistically significant*. Therefore, you might (or you should) consider a cross-validation approach to whether or not the variable is truly useful for predicting the price well, rather than p-values or confidence intervals.
- To judge overall model fit, the text (and 6530) discuss many different approaches. Remember that measures like R-squared and comparison of MSE fall short in comparing models within a given set of data. This is because a more complex model will always out-perform a less complex model on data it can 'see.' Commonly a comparison of full vs. reduced model can be used on a particular dataset, which is another way of looking at F-tests. In this test, you can compare two models with different numbers of predictor variables.
- The goal is that a model should out perform other models on data it has not seen before. This is why cross-validation (training, test set logic) is a useful measure for judging model fit. We want our model to out perform the other models on the test data.
- Logistic regression is a common approach to modeling many 'real world' situations; however, there are many more algorithms that have the ability

to predict better than linear regression. If you are interested in interpretability, this is a great algorithm. If you think you are going to find a model that predicts an outcome with better accuracy than any man ever has done before, this modeling technique should not a be a first choice for most 'real world' situations.