STAT 213: Project 2 (Logistic Regression)

Colin Reimer Dawson

Due via Blackboard, Friday May 4th by 11:59 P.M.

This project is almost exactly analogous to project 1, except you will use logistic regression in place of linear regression. That is, your goal is to create a predictive model in a domain of your choice, where the response variable is binary, and the prediction is the probability that one of the two outcomes occurs. You may use any of the tools and methods you have learned about in class, *with the exception of automated model selection methods* (more on this below).

Choice of Topic

Some examples of topics are below, though I encourage you to pick your own topic in an area that is of interest to you. There are some links to repositories where you can find datasets, arranged (most) by domain, on the course website under "Resources" (if you're reading this electronically click here)

As with previous projects, you will document the process and write about the results in an RMarkdown document.

Here are a few examples of the kinds of things that might make good topics:

1. Modeling the 2016 presidential election on a county-level (i.e., try to predict the probability of a Trump plurality vote using other county-level variables)
2. Predicting the probability that a baby's birthweight is below some medically significant threshold using variables related to the mother, the pregnancy, etc. (variables related to the father, the household, etc. might be relevant too, if they're available).

3. Build a model to predict whether a tumor is malignant using various measurements taken using imaging. A pre-split training set, test set, and some documentation are available here:

   Training Set: `http://colindawson.net/data/cancer_train.csv`

   Test Set: `http://colindawson.net/data/cancer_test.csv`

   Documentation: `http://colindawson.net/data/cancer-codebook.txt`

   Note that the predictor variables in this data are integers from 1 to 9, and it is not always clear whether these should be treated as quantitative or categorical.

When choosing your dataset, start with the cases and response variable that you want to be able to model (the response must be binary, though you can dichotomize a quantitative response if it makes sense to do so), and try to find data that contains information about that response variable for a sample of cases, as well as several other potentially useful predictor variables. You can gather the data yourself if you want to, but if you do this you should talk to me about your plans first before carrying out any data collection.

An Outline of the Modeling Process

You can make use of any metrics, tools, code, etc. that we have used in class, except that this time you should not rely on automated selection techniques. The reason for this is that I want you to focus on individual model comparisons and interpretation.

There is not a set path of model-fitting and assessing that you must follow, but here is a basic set of elements to include. Note that this is slightly different from what was written for Project 1, both due to the fact that you are not using automated techniques, and because I wanted to provide a bit more structured guidance.

1. Before fitting any models, set aside a random subset of your dataset (perhaps 20% of cases) as a "test set" that you will use only to evaluate prediction error for your handful of "finalist" models. It should not be used for any parameter fitting or hypothesis tests. *In order that your results are reproducible, and that the random split doesn't change every time you run your code, remember to* `set.seed(SOMENUMBER)` *at the start of your Markdown.*

2. Check the distributions of your variables individually to see whether any of them clearly should be transformed, before fitting models. Fit single predictor

models and construct binned residual plots to get a sense whether polynomials could be needed.

3. Check for multicollinearity among a set of (possibly transformed) predictors (without higher powers), to see whether any variables should be removed entirely, or whether you might want to combine multiple variables into one (for example, in a country-level dataset, GDP and population are highly correlated; it might make more sense to take their ratio, which is interpretable as GDP per capita).

4. Use your background knowledge together with any appropriate statistical methods you know about to build a handful of models, *fitting them only on the training set*, that allow you to address interesting questions about your response variable. Check for possible outliers by plotting deviance/Pearson residuals.

5. Use nested tests and AIC as a guide to whether to include polynomial terms, interactions, or add and remove predictors, but don't take either the $P < 0.05$ threshold or the AIC score as gospel; if a comparison yields a borderline result, you might want to consider both models. As you are doing this, *describe in text what it means in context to choose one model over another.*

6. After you have done some winnowing of the pool of models, compute some k-fold cross-validation metrics for those models *within the training set.* This will provide you with another measure (in addition to your nested tests and AIC measures) of model quality. If any models in your pool seem clearly worse on all measures, you might remove them at this stage.

7. For each of your (possibly revised) finalists, compute some measures of generalization performance on the test set. As before, note that you should *not* re-estimate your coefficients on the test set; use the coefficients that were fitted on the 80% of cases in the "training set" to make predictions for the test set, comparing the predictions to reality. For each measure of generalization performance, compute the same thing on the training set and compare the results (making sure to do any appropriate adjustments for sample size if the measure is not already adjusted) to assess the degree of overfitting that occurred.

8. Once you have settled on a final model, report and interpret coefficients, and calculate, report and interpret confidence intervals *on the probability scale* at a couple of representative combinations of predictors. Since you have already settled on this as your final model, you should *not* do hypothesis tests for the coefficients at this point.

RMarkdown Notes and Tips

You should turn in a Markdown report documenting your data exploration and modeling process. Please note:

- The text and code should be part of a single `.Rmd` document; don't cut and paste into a separate document, as this introduces the possibility that your report becomes out of sync with your data and code!
- Don't wait until you're done to attempt to Knit your document! If you do this it will make it harder to track down the sources of errors that you get when Knitting. Remember that the `.Rmd` must be completely self-contained, and include any commands that load packages, import data, etc. If your code runs chunk by chunk but will not Knit, the most common reason is that you read in your data or loaded a package with the drop down menu or a check box, which makes it available in your interactive session, but does not make it available when Knitting, instead of with a command in your document, which does both. The problems with Knitting directly to PDF on RStudioPro should be fixed.
- When reading in a `.csv` file, put it in the same directory as your `.Rmd` and write your `read.file()` command in the form `read.file("MyDataSet.csv")`, not `read.file("~/path/to/MyDataSet.csv")` This way it will be portable and Knit on someone else's machine (most notably mine), as long as the data file is in the same directory as your `.Rmd`.

An Outline of the Writeup

The writeup should consist of:

1. A brief introduction of why your chosen topic is of interest, in general, and to you specifically.
2. Since this is a methodology project, the technical "meat" of the writeup is the "Methods and Results" section, which walks the reader through your modeling process, describing the logic behind each choice that you make (with any appropriate supporting numerical or graphical justifications).
3. A "Discussion" section in which you interpret your findings (both qualitatively and quantitatively) in context

4. If there are results you want to report that would be cumbersome to include in your main narrative, put them in a section at the end labeled `Appendix`.

Upload both the `.Rmd` source and the compiled `.pdf` output as usual, and if the data is in a local file as opposed to a URL, the data set as a `.csv`).

In the interest of clean presentation of results, before submitting your final document, add the following snippet in your first code chunk.

```
opts_chunk$set(
    echo = FALSE,
    message = FALSE,
    results = 'hide'
)
```

This will suppress code, unwanted messages, and text output from the final Knitted document. Note that this means that you cannot rely on raw R output to convey your results: you will need to refer to key results in the text. For numerical results you want to report in the text, store the value in a variable and use the `‘r variable_name‘` syntax in your text to include the value of `variable_name‘` in a sentence.

For your Appendix section, if you have one, you may want to change these options; for example, setting `results = 'markdown'` to display the raw output.

Grading

The project is graded on the following "Content" SLOs:

1. F2: Demonstrate that you can convert between probability, odds, and log odds and understand the differences
2. F3: Distinguish the effect of changes to a predictor on probability vs. odds vs. log odds
3. F4: Interpret tests of coefficients in logistic regression models.
4. F5: Correctly interpret confidence intervals for the predicted probability in logistic models
5. E2/F6: Identify and employ appropriate nested model comparisons to answer substantive questions

6. E5/F7: Sensibly employ metrics such as AIC, and cross-validation prediction error, to select among competing models

In addition, the project is graded on the following "holistic" criteria (all weighted equally, on a 0-10 scale):

1. Overall technical soundness: To what extent are the tools you apply appropriate to the questions you are trying to answer, and to what extent are the technical details correctly executed in the code?

2. Chain of reasoning: How well are the decisions you make throughout your analysis motivated in terms of the research question and in terms of the results obtained so far?

3. Interpretation of results in context: How well did you take the results of your analysis and connect it back to the real world context of the problem, in such a way that a reader not trained in statistics can take something away from your analysis?

4. Clarity of communication: How easy is it to follow your writeup? This criterion comprises both the quality of the writing itself, the organization of the report (are text sections, graphs, etc. well placed for the reader to follow what is going on?), and the aesthetic quality of the report (have you suppressed unnecessary/distracting output, are your figures well labeled and visually appealing?)