LOGISTIC REGRESSION II

Paul Speaker

Train/Test Splits

- Standard practice for modeling is to train a model on part of dataset and test on rest of data
  - Allows view of how model will do on data that model was not optimized for
  - Avoids overfitting
- How much?
  - Some variation in opinion, but anything 70-80 : 30-20 is typical
  - For smaller dataset test dataset should be higher proportion (maybe even 50/50)
- Want to have split be random
  - Can use sample function in R (using sample to choose random row numbers for split)
- Want performance to be similar between train and test datasets
  - Create confusion matrix for each, compare results

The Predict Function

- How to test model—the predict function in R
- Syntax: predict(<model object>, <test dataframe>, type = ‘response’)
  - Output is the set of predictions
  - Type = ‘response’ ensures that output is a probability, $P(Y = 1|X)$ as opposed to odds ratio or argument for exponential in logistic function
- Then output can be fed into the ConfusionMatrix function like original fitted values
  - Need behavior between train and test to be relatively close to each other

Calibration

- Calibration allows one to test whether results can in fact be interpreted as a probability
  - Idea:
    - Bin the results based on fitted values (often in deciles)
    - Take for example the 50%-60% bin
      - Across the bin, is it true that 50-60% of that bin are in fact <target = 1> cases?
    - Can test both train and sample datasets (or just overall)
    - Doesn’t have to be exact (particularly with smaller datasets)
      - This is especially true “in the middle deciles”
- Suppose result is not calibrated. What to do then?
  - Could adjust probabilities to match true probabilities observed
  - Try another model (probably should do so anyway!)

Bootstrapping Overview

- Bootstrapping is a powerful method to perform inference on ML parameters based on resampling
- Nonparametric*
  - Especially powerful when there is no reason to assume particular form for underlying distribution
- Allows for the construction of confidence intervals in general way regardless of sample statistic
- Resampling done with replacement

*Still requires assumptions of Central Limit Theorem to Hold

Bootstrapped CI's for Parameters

- The bootstrap method can be used to produce confidence intervals for the parameters of logistic regression, just like with linear regression
  - Can be either regular or Bayesian bootstrap (will not cover Bayesian bootstrap here)
- We sample (with replacement) rows from a dataframe rather than from a single field
- Pseudocode
  - N <- length(df)
  - for(i in 1:<large-ish number>) {
    - Boot_df <- sample(<rows in data frame>, N, replace = TRUE)
    - Coeff[i] <- glm(<model form in Boot_df>)$coefficients }
  - <CI = quantiles of coefficients for all different models>

Contour Plots

- Contour plots can give a lot of intuition to classification models
- Limitation: works best with 2 numerical values (can facet by categorical variables)
- Each curve represents constant probability level
- For logistic regression, result is always set of parallel lines
- Not equally spaced (closer for middle values