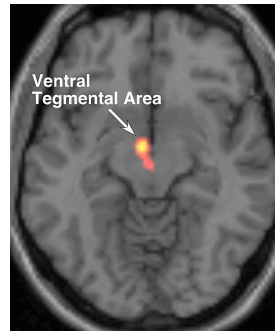


Lecture 10. Future Methods, Wrapping Up

Questions from Previous Classes

- Overfitting in the news
Brain imaging, fMRI scans



<http://www.npr.org/templates/story/story.php?storyId=106235924&ft=1&f=2>

Topics

Comparison of methods using ANES

Hybrid methods

Model averaging

Modeling text

Data Preparations

anes_2008, anes_2008_voters

- Revised yet again!
- Latest version has missing values fixed up so we can get consistent results.

Numerical variables

- Filled in missing values in columns of ANES 2008 with means
- Added a dummy variable to show which cases were filled (as a categorical variable)

Categorical variables

- Filled in missing cases (blank text) with "Missing"
- Otherwise left categorical column as is (did not expand into lots of indicators)

Contest

Data

- Estimate a model based on sample of 1,000 voters. In anes_2008_voter_sample.jmp
- Predict the other 603 voters. All are in anes_2008_voters.jmp
- Cannot use the post-election FT values

Objective

- Scoring rule is based on deviation of predicted probability of voting for Obama compared to observed behavior (1 if vote for Obama, 0 otherwise).
- Not the same as the classification error rate which would count the number correct/incorrect, regardless how close you come to the right prediction.

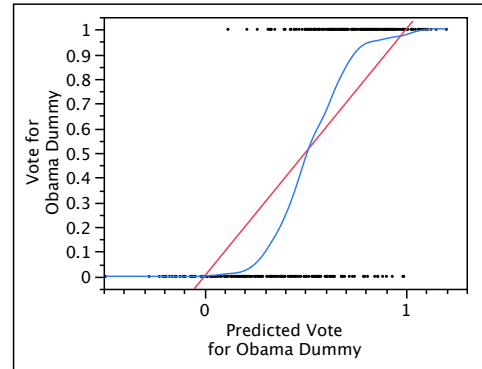
Methods

- Stepwise regression using Bonferroni
 - Option to add a few more if many come into the fit. (i.e., increase the p to enter from $0.05/(\text{number possible})$ to $k/(\text{number possible})$ where k is the number that are added based on the original.
 - Add a calibration variable as needed. No need for cross-validation.
 - Stepwise suffers: (1) it works better with the split-up categorical dummy variables rather than leaving the categorical variables as “bundles.” (2) Using a 0/1 response.
- Logistic regression, piggy-backed on the regression
 - Let the regression pick the right sort of variables, perhaps a few beyond the cut off. Then retain only those that are interesting (i.e., large t -statistics)
- Neural network
 - Lots of variables, but guided by the prior regression models (cheating a little)
 - Split-sample cross-validation
- Classification tree
 - All of the variables, using split-sample cross-validation
- Hybrid
 - Use the predictions from neural network, classification tree and logistic regression as 3 leading variables in a stepwise model fit to the full data set (since no need to validate).
- Details
 - All models fit using the 1,000 estimation cases.
 - Save the prediction formulas for each model.
 - Save the dialog expressions used to build the models in the jmp data table.

Stepwise Regression

Baseline for comparison

- Run without interactions “manually” using the 125 supplied explanatory variables (Region..Torture)
- Hard to count variables since the categorical items are bundled.
- Get at least 9 explanatory variables (mangled due to JMPs representation of categorical)
- $R^2 \approx 0.65$, but not well calibrated. Calibrated predictor improves R^2 to about 72%.
- Predictors “make sense” but with collinearity, its easy to read into these too much meaning. One of these is a missing data indicator.



Interactions

- Tried adding just a few of these... but a better tool/interface would make this easier.
- Obtain similar R^2 and model (correlation of predictions is about 0.99)

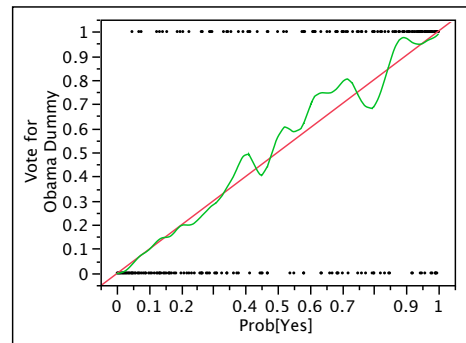
Logistic Regression

Initial model

- Too many variables to fully search using stepwise logistic (I tried by accident)
- Start with the model identified by OLS, then swap the response to the categorical variable.
- Most of the variables are significant but for the missing indicator (too few cases)

Calibration

- The logistic fit is much better calibrated than the linear fit, without need for further calibration.



Neural Network

Switch dataset

- Need to use the file with just the test sample `anes_2008_voters_sample`
- Otherwise cross-validation sees the results for the test sample.
- Split sample for cross-validation (with 500 reserved for testing)

Which variables?

- Use the form of predictors suggested by the stepwise search.
 - Race
 - Pre-election FTs along with their missing indicators
 - Party ID
 - Hopeful-Afraid of Obama
 - Crooks in govt
 - Handling Iraq war
- Slip in a few others (Region..Gender)

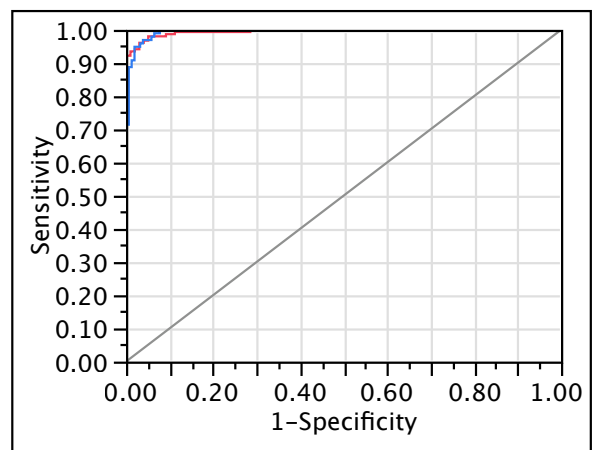
Which fit?

- Picked a slightly over-fit model (with penalty 1) that has a best of these fits out of sample.

Fit History				
Nodes	Penalty	RSquare	CV RSquare	.2 .4 .6 .8
3	1	0.79128	0.64055	
3	2	0.70079	0.61612	
3	4	0.58637	0.55180	
3	0.5	0.86774	0.63380	

Claimed performance

- ROC curve for this model (must be an in-sample estimate) is stunning, with an AUC of 0.995.
- Estimate the model using the full data set in order to get better parameter estimates. When fit to the full sample, claims R^2 at 81%, but that's an in-sample estimate.



Classification Tree

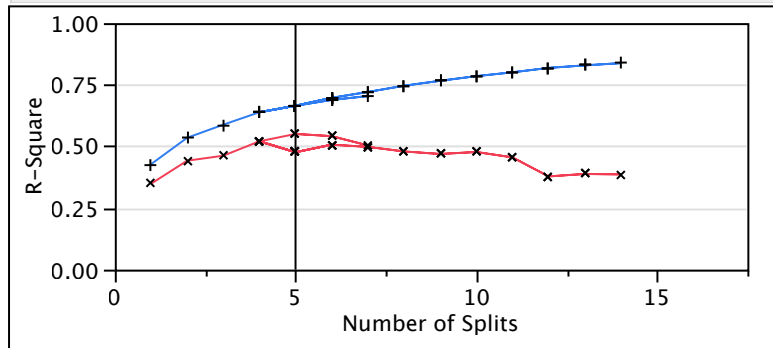
Which variables

- Don't rely on any of the prior results, just pick them all! (Region..Torture)

Results

- Split-sample CV, using the "Go" button
- Only 4 splits before out-of-sample accuracy falls.
- Do slightly better with a minimum node size set to 20

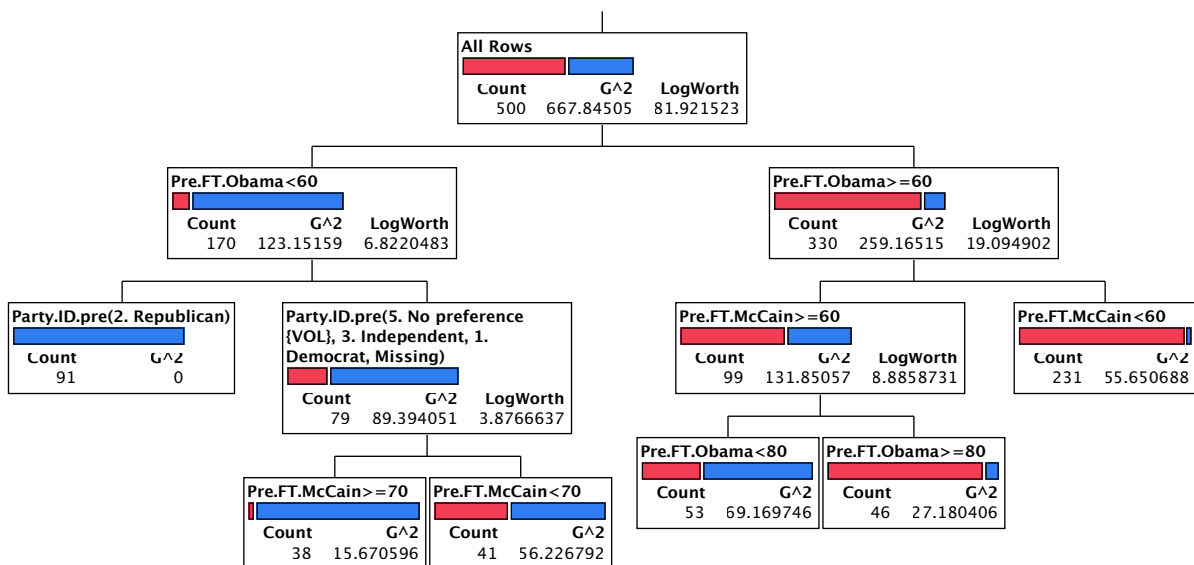
Split History



Excluded in Red

Chosen tree is familiar

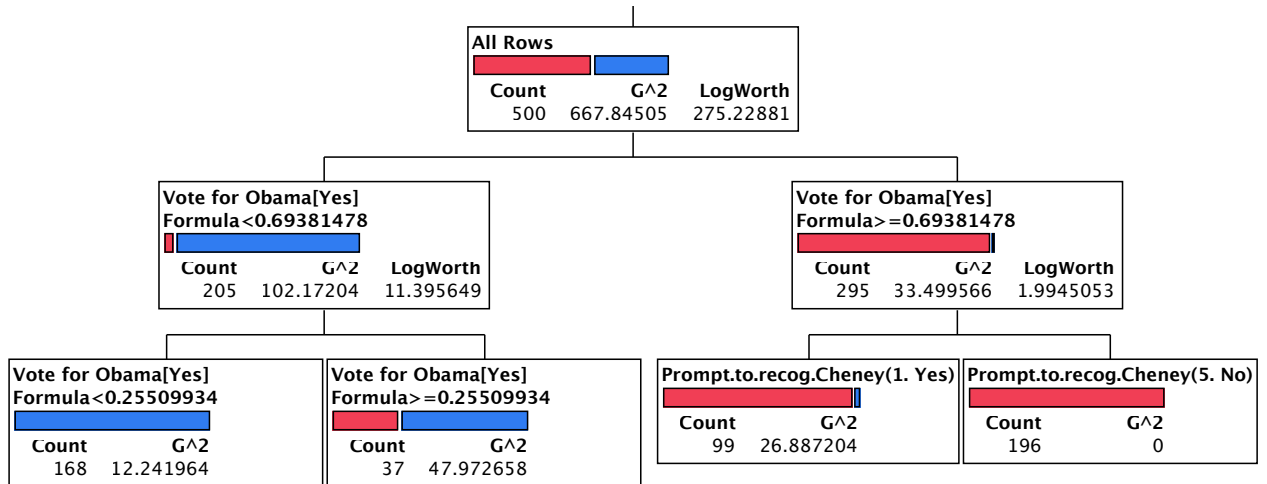
- Dominated by pre-election FTs with Party ID tossed in.
- Suspect that the tree will be too discrete to predict well.



Hybrid Tree

Use other predictions

- Combine the tree with the neural net, using its predictions as well as the hidden notes.
- Trees are invariant of transformations of the variables, so its OK if the variable has an odd distribution; the tree only uses the ordering of the cases.
- No particular improvement in the fit, though it is interesting to see Cheney in the fit.



Comparison

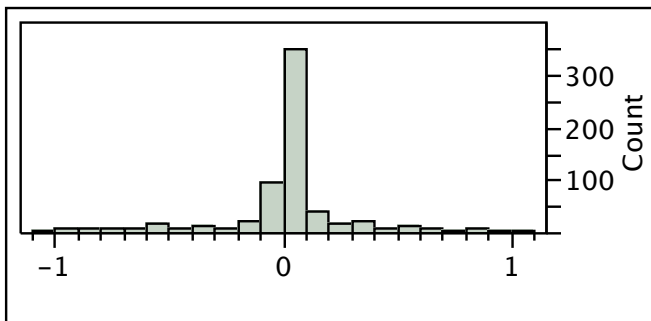
Glue the formulas together

- Need to move those from the smaller subset to the full data table.
- Then build formulas for the prediction error

And the winner is...

- No surprise. Nothing does particularly better than calibrated stepwise regression.
- Comparison of the errors between in-sample and out-of-sample indicate little over-fitting in any of the models.

Calibrated Error



Moments

Mean	0.0151234
Std Dev	0.2489999
Std Err Mean	0.0101401
Upper 95% Mean	0.0350376
Lower 95% Mean	-0.004791
N	603

Model Averaging

Better use of data (CV)

- Split-sample cross-validation introduces randomness into the results
- Analysis depends on the sample that we happen to get for estimation and validation.
- Rather than do one split, do several.

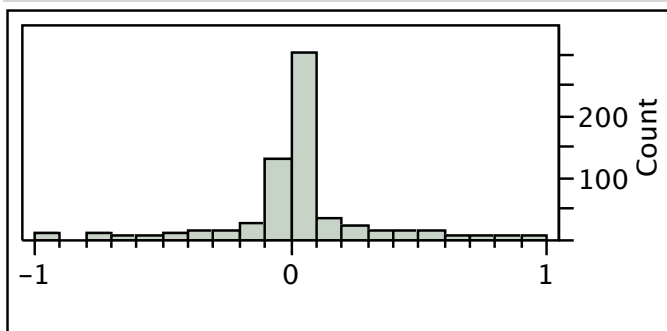
What's the predictor?

- Average the predictions from the various models
- Use a regression model

Employ the same idea here...

- Have several predictors of election outcome
calibrated regression, logistic regression, neural network, tree
- Combine by averaging

Average Predictor Error



Moments

Mean	0.0186332
Std Dev	0.2399545
Std Err Mean	0.0097717
Upper 95% Mean	0.037824
Lower 95% Mean	-0.000558
N	603

- Results from averaging are very slightly better than an individual model
- We'd have gotten better performance by averaging had we not let the stepwise model leak itself into the other models

Key terms for model averaging

- Shows up in Bayesian modeling (different from what we're doing)
- Boosting
Refit the model repeatedly, putting more weight on the cases that were missed
- Bagging
Estimate the model on "bootstrap samples" from the data, then average the collection of predictions from the different models
- Specific technique: random forests, claimed to be the "best off-the-shelf" classifier around.

Text Mining

Examples

- Identifying similar documents (sorting, classification)
- Grading written essays, as in the new SATs
- Identifying plagiarism

Methods

- Bag of words to represent a document, sometimes bi-grams or tri-grams.
- Correlations of sparse vectors
- Often unsupervised (i.e., more like clustering than classification)

Different problem

- What does a word mean?
- Simplified version of the problem
What's a proper noun mean in a specific context?
- Examples: Indianapolis... city, race, ship
Georgia... person, state of US, country in Asia

Wiki project (with Dean Foster)

- Exploit the Wiki disambiguation index that provides a “defining” page.
- Develop regression models that predict classification
- Trick is to build a regression data set and good predictors

Building the data

- Pick proper noun “Indianapolis” and its definitions.
- Identify collection of all pages that use the word Indianapolis in each sense.
- Random selection steps
 - a. Pick two definitions of Indianapolis at random, e.g. the race and the city.
 - b. Pick a source page at random (50/50 chance) from among those that point to one definition or the other. Suppose we get a page that points describes the race.
 - c. Pick one of the definitions at random. Suppose we pick the page that defines the City of Indianapolis.
- Define a 0/1 response depending on whether the page chosen points to the selected definition. In this example $y = 0$ since the definition and page don't match.
- Build whatever predictors you want from the source page and the defining page.

Model building

- Streaming (sequential) feature selection using stepwise regression as described (way too briefly) in Lecture 6.
- Needs to be streaming since there's a potentially infinite collection of predictors.
- Build the features as the regression modeling proceeds, using results from the success of prior predictors to decide which features to build for further modeling.

Bigger task... definitions of any words**Current state**

- Let's get a grant!