

Wrap-Up

Dave Glidden
1 Dec 2009

Theme of Class

- Standard regression:
 - assumes pre-specified model
- Typical problem: build “best” model
 - best is defined in data-driven way
- Point estimates biased, test size > 0.05
- Model overfitting
- Model optimism

Overfitting

- Searching the data for patterns
 - drawn to most extreme
- Leads to regression to mean
 - most extreme feature not replicable
- Most extreme: false positive findings

15/17 Coefficients lower in Validation Set

Variable	Training Coef.	Valid Coef.	Ratio
Age 60-64	0.63	0.43	0.68
Age 65-69	1.04	0.81	0.78
Age 70-74	1.31	0.95	0.72
Age 75-79	1.69	1.41	0.84
Age 80-84	2.12	2.03	0.96
Age > 85	2.79	2.68	0.96
Male	0.71	0.77	1.09
Diabetes	0.58	0.57	0.98
Cancer	0.72	0.51	0.71
Lung Disease	0.82	0.46	0.56
Heart Failure	0.85	0.76	0.89
BMI < 25	0.50	0.55	1.10
Smoking	0.72	0.47	0.65
Bathing	0.67	0.39	0.58
Money	0.64	0.35	0.54
Walking	0.74	0.69	0.94
Pushing	0.42	0.40	0.94

Regression to Mean

- Overfitting is version of regression to mean
- Starts with selecting bests: best functional form, best predictor, best cut-point
- Unknown signal/noise ratio
- Noise may predominate & not replicate

Signal/Noise Ratio

- Overfitting is extreme when
 - sample size is small
 - many predictors examined
 - large % non-significant predictors
 - many hypotheses tested
- Varies by dataset!

Optimism

- Models selected to optimize criteria
 - high likelihood, low sum of squares
- No account of data-driven hypotheses
- More predictors => better model
- Model looks too good on data which developed it
- Want effects that reflect truth outside data

Good to be data-driven

- Nothing wrong with data-driven questions
 - answers not always reliable
 - especially if you ask a lot
 - and if data is small
- Data's sturdiness varies
 - big n, can tolerate more questions

Is Effect Real?

- \$1,000,000 question
 - is there any other question in statistics?
- Can't know without replication
 - few people have replication to spare
- Can't assess hypothesis
 - can assess the strategy
- Powerful tool: cross-validation, bootstrap

10-fold Cross-Validation

- Create 10* random subsets of the data
- In turn...
 - build a model on 90% of data
 - assess predictive performance of 10%
 - until cycled thru data
- Every observation gets predicted
 - by some independent data

Cross-Validation

- Good strategy for assessing strategy
- How data aggressive to make a method
 - how many predictors to pick
 - or what p-value cut off for stepwise?
- Requires some kind of automatic method
- Requires a criterion to be optimized
- Needn't be 10-fold, but generally best

Pros/Cons

- P: unbiased estimate of prediction error
- C: nearly unbiased (lose 10% of data)
- P: this is actually conservative
- C: doesn't give exact pred. error (estimate)
- C: works best for automatic strategy

Bootstrap

- Another powerful resampling technique
- Drawn samples of n observations with replacement
- Same observation can appear >1 time
- Some observations left out

Bootstrap

Data ID	Number of Times Sampled			
	BS #1	BS #2	BS #3	BS #4
1	0	0	0	2
2	2	1	0	1
3	1	1	1	0
4	0	2	3	0
5	2	1	0	2
6	2	1	1	2
7	1	1	2	0
8	0	1	1	1

Optimism Correction

- Build a model on the bootstrap sample
 - select predictions, get coefficients
- Apply these models to original dataset
- Will correct for optimism performance
- Not easy to implement in Stata
 - cross-validation is easy, better

Stepwise Stability

- Can use bootstrap to investigate stepwise model
- Bootstrap data, apply stepwise selection
- See how often predictor is selected
- Gives a sense of ranking of variables

AIC/BIC

- A measure of model performance
- Based on log-likelihood
measure of explained variation
- Correct for model optimism
more predictors always reduces variation
- BIC: greater penalty for more predictors
- Have analogy to p-value cut off
AIC: $p < 0.16$, BIC: p depends on n

These Tools

- All have some ability to correct for optimism
- Helpful in model building or model assessment
- Bootstrap and CV are most flexible
computational issues based on randomness
- AIC/BIC easily computed

Homework #3

- Primary Biliary Cirrhosis (n=244)
- Outcome: Death in 5 years
- Series of 7 predictors
excluded some cts ones, use them here
- Build a model based on data

How to...

- How to choose/rank variables?
base on predictive power
- How large to make model ?
- How to handle continuous variables ?
- How to make a score ?

CV C-statistic Strategy

- Considerations: external prediction
best assessed with CV on C-statistic
- Choose smallest model with good prediction (CV C-statistic)
- Rank variables by stepwise selection
bootstrap is very helpful here

Bootstrap Selection

- Bootstrap data, say, 1,000 times
- Apply stepwise model (backwd, 0.05 level)
- See which variables selected
- Repeat for all bootstrap data

Stata Command

Backward 0.05 level selection

```
xi: bootstrap, saving(bs.dta, replace) reps(1000):  
stepwise, pr(0.05) : logistic dead5 sex asictes  
hepatom spiders edema bilir copper alk sgot age
```

Read in the bootstrap data

```
use bs.dta, clear
```

Results

```
. summ
```

Variable	Obs	Mean	Std. Dev.	Min	Max
__b_sgot	534	.0150777	.005349	.0061012	.040295
__b_asictes	350	2.492184	.6377221	1.245005	6.109618
__b_hepatom	801	1.620955	.4974126	.8386332	4.017365
__b_spiders	473	1.437389	.3987586	.8425309	3.964958
__b_age	982	.1103722	.0340895	.0404368	.2506711
__b_bilirub	966	.4512025	.2290442	.106947	1.681895
__b_copper	913	.0114224	.004253	.0043968	.0349195

look how often selected

Backward Selection

Variable	# Times (%) Selected	
	p < 0.05	p < 0.10
Age	982 (98%)	998 (100%)
Bilirubin	966 (97%)	986 (99%)
Copper	913 (91%)	944 (94%)
Hepatomegaly	801 (80%)	881 (88%)
SGOT	534 (53%)	628 (63%)
Spiders	473 (47%)	573 (57%)
Asictes	350 (35%)	416 (42%)
Edema	0	0
Alk. Phos.	0	0

Forward Selection

Variable	# Times (%) Selected	
	p < 0.05	p < 0.10
Age	982 (98%)	994 (99%)
Bilirubin	981 (98%)	987 (99%)
Copper	911 (91%)	939 (94%)
Hepatomegaly	786 (79%)	870 (87%)
SGOT	524 (52%)	653 (65%)
Spiders	0	0
Asictes	0	0
Edema	472 (47%)	513 (51%)
Alk. Phos.	0	493 (49%)

AIC/BIC/ AUC Table

# Pred	Var	AIC	BIC	CV AUC
1	+Age	303.8	310.8	0.63
2	+Bili	204.6	215.1	0.88
3	+Copper	189.1	203.1	0.89
4	+Hepat	182.2	199.6	0.90
5	+SGOT	180.7	201.6	0.89

AIC wants even more variables

My synthesis

- Four variable model looks best for pred.
- Will not consider > 4 variables
poorer AUC
bootstrap doesn't clearly rank beyond 5
- Top 3 variables are continuous
going to consume df when categorized
makes me wary about hepatomegaly
- My choice: age, bilirubin and copper
many reasonable choices here

Continuous Variables

- How to select cutpoints?
- No good way, just avoid a bad way
- Try to see if 2, 3, 4 categories better
- Use percentiles
- Automatic selection based on CV AUC

Optimizing Categories

Categories	AIC	BIC	CV AUC*
None/Cts	189.13	203.09	0.89
2	211.50	225.45	0.83
3	198.43	222.85	0.87
4	191.72	226.61	0.88
5	184.67	230.03	0.89

* Cross-Validated Area Under the Curve

Selecting Categories

- Two seems to under predict
- Three seems like a lot of cutpoints
*6 degree of freedom model
larger than I'd like*
- Balance prediction, simplicity, interpretation

Cutpoints

```
centile age bilir copper, c(33 67)
```

Variable	Obs	Percentile	Centile	-- Binom. Interp. -- [95% Conf. Interval]	
age	244	33	45.79302	44.09498	47.36089
		67	55.43847	53.5342	56.41853
bilirub	244	33	.9	.8	1.1
		67	2.815	2.1	3.4
copper	242	33	50.19	42.1802	57
		67	105	86.28326	123

consider rounding these
age: is age of 45 v. 55 really important?

Recoding Variables

```
recode age min/45=0 45/55=1 55/max=2, gen(cut_a)
(244 differences between age and cut_a)
```

```
recode bilir min/1.0=0 1.0/3.0=1 3.0/max=2, gen(cut_b)
(244 differences between bilirub and cut_b)
```

```
recode copper min/50=0 50/100=1 100/max=2, gen(cut_c)
(242 differences between copper and cut_c)
```

Leading Model

odds ratio

```
xi: logistic dead5 i.cut_b i.cut_a i.cut_c
i.cut_b      _Icut_b_0-2      (naturally coded; _Icut_b_0 omitted)
i.cut_a      _Icut_a_0-2      (naturally coded; _Icut_a_0 omitted)
i.cut_c      _Icut_c_0-2      (naturally coded; _Icut_c_0 omitted)
```

```
Logistic regression                                Number of obs   =      242
                                                    LR chi2(6)      =     128.18
                                                    Prob > chi2     =      0.0000
Log likelihood = -92.155717                       Pseudo R2      =      0.4102
```

dead5yr	Odds Ratio	Std. Err.	z	P> z	[95% Conf. Interval]
_Icut_b_1	5.000409	2.733131	2.94	0.003	1.712989 14.59676
_Icut_b_2	37.35786	22.56645	5.99	0.000	11.43415 122.0563
_Icut_a_1	2.594782	1.270439	1.95	0.051	.9938978 6.774233
_Icut_a_2	10.25026	5.468493	4.36	0.000	3.60264 29.16414
_Icut_c_1	3.092849	1.703814	2.05	0.040	1.050611 9.10491
_Icut_c_2	6.997907	3.888321	3.50	0.000	2.355082 20.79363

How do I get those points?

Leading Model coefficients

```
xi: logistic dead5 i.cut_b i.cut_a i.cut_c, coef
i.cut_b      _Icut_b_0-2      (naturally coded; _Icut_b_0 omitted)
i.cut_a      _Icut_a_0-2      (naturally coded; _Icut_a_0 omitted)
i.cut_c      _Icut_c_0-2      (naturally coded; _Icut_c_0 omitted)

Logistic regression                               Number of obs   =       242
                                                    LR chi2(6)      =       128.18
                                                    Prob > chi2     =       0.0000
Log likelihood = -92.155717                       Pseudo R2      =       0.4102
```

dead5yr	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
_Icut_b_1	1.60952	.5465814	2.94	0.003	.5382398	2.6808
_Icut_b_2	3.620543	.6040617	5.99	0.000	2.436604	4.804482
_Icut_a_1	.9535026	.4896128	1.95	0.051	-.0061209	1.913126
_Icut_a_2	2.327303	.5334978	4.36	0.000	1.281667	3.37294
_Icut_c_1	1.129093	.5508882	2.05	0.040	.0493718	2.208814
_Icut_c_2	1.945611	.5556406	3.50	0.000	.8565756	3.034647
_cons	-4.944819	.7273103	-6.80	0.000	-6.370321	-3.519318

Deriving the Points

Variable	Coef.	Coef/Min (Coef)	Points
Bilirubin 1-3	1.6	1.7	+2
Bilirubin > 3	3.6	3.8	+4
Age 45-55	0.95	1.0	+1
Age > 55	2.32	2.4	+2
Copper 50-100	1.12	1.2	+1
Copper > 100	1.94	2.0	+2

age 45-55 has smallest coefficient

Runner Up #1

Logistic regression

Number of obs = 242

LR chi2(5) = 120.74

Prob > chi2 = 0.0000

Pseudo R2 = 0.3864

Log likelihood = -95.875666

dead5yr	Odds Ratio	Std. Err.	z	P> z	[95% Conf. Interval]
_Icut_b_1	4.92575	2.598672	3.02	0.003	1.751479 13.85287
_Icut_b_2	29.27064	16.63652	5.94	0.000	9.608061 89.17204
_Icut_a_1	4.476291	1.766286	3.80	0.000	2.065595 9.700439
_Icut_c_1	2.978259	1.579461	2.06	0.040	1.053281 8.421325
_Icut_c_2	6.770815	3.600917	3.60	0.000	2.38752 19.20148

age cut at 50 years old

Points for 2nd Model

Variable	Coef.	Coef/Min (Coef)	Points
Bilirubin 1-3	1.6	1.5	+1
Bilirubin > 3	3.4	3.1	+3
Age > 50	1.5	1.4	+1
Copper 50-100	1.1	1.0	+1
Copper > 100	1.9	1.7	+2

2nd Runner Up

Variable	Coef.	Coef/Min (Coef)	Points
Bilirubin 1-3	1.4	1.4	+1
Bilirubin > 3	3.1	3.0	+3
Age > 50	1.5	1.4	+1
Copper 50-100	1.1	1.0	+1
Copper > 100	1.9	1.9	+2
Hepatomegaly	1.1	1.0	+1

Final Considerations

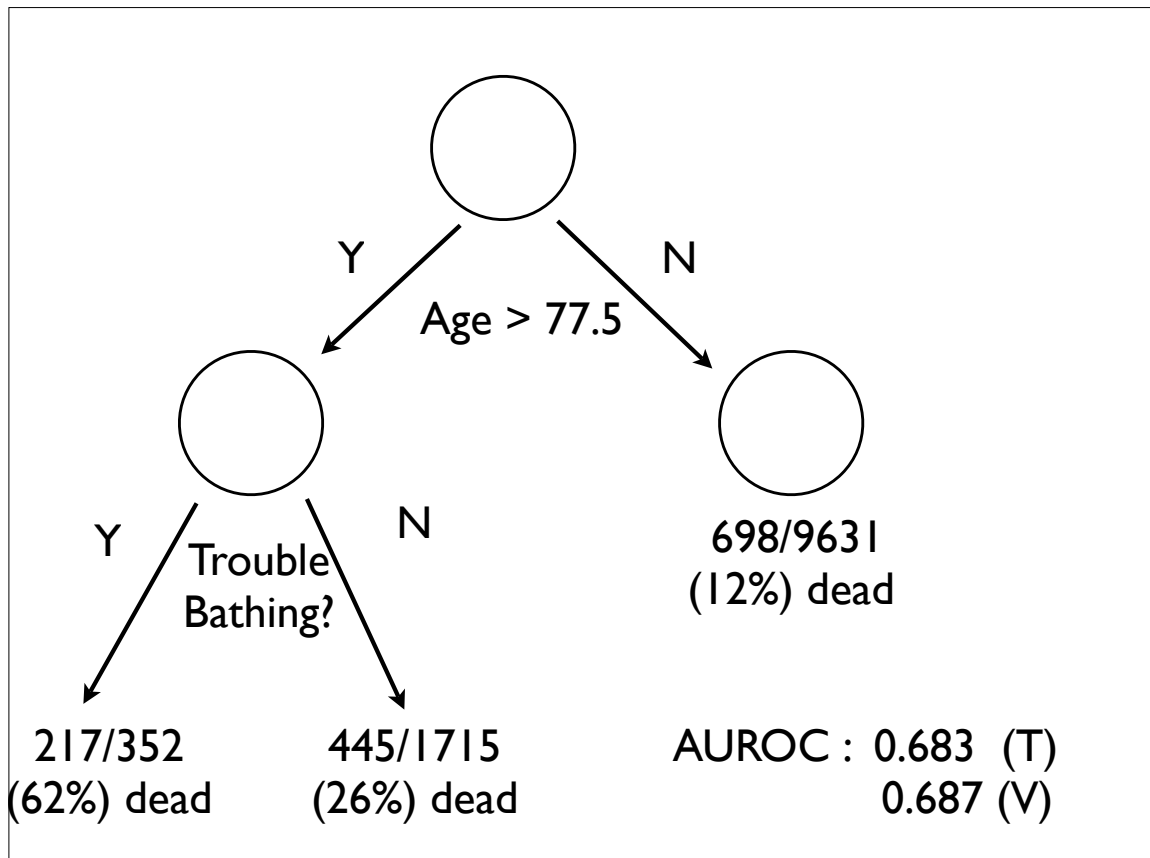
- All 3 models similar statistical properties
area under ROC not so different
- Weigh their practical value
- Many other approaches to constructing model

Final Model

- Embodies many choices/compromises
- Not perfect model
- Avoid a bad model
- Thoughtful choices
- Track runner-up models
- Logistic regression: one strategy

CART

- Classification and Regression Trees
- Creates model based on splits of data
- Splits selected for maximum statistical significance
- Produces nodes with similar prognosis
- Consider application to Lee Data



CART

- Automated method
- Data hungry: make decisions based on maximizing data split
- Most important split is first
- Cross-validation reins in overfitting

Appealing Features

- Mimics decision making
- Interpretable classifications
- Cuts for continuous variables
- Automatic detection of interactions
- Non-computation approach to grouping

Unappealing Features

- Not available in Stata
- Different in spirit from regression
 - no p-values, coefficients
- User controls awkward
- May split on age many times
 - inefficient if covariate really additive

Machine Learning

- Last decade has seen proliferation
- Automatic techniques: SVF, bagging, random forests
- All produce excellent classifications
- Not easy to interpret
- More black box in their nature

Evaluation

<http://psg-mac43.ucsf.edu/ticr/CourseEvaluations/ticreval.asp?id=362>

extremely helpful for a new class