Notes on Measuring the Performance of a Binary Classifier

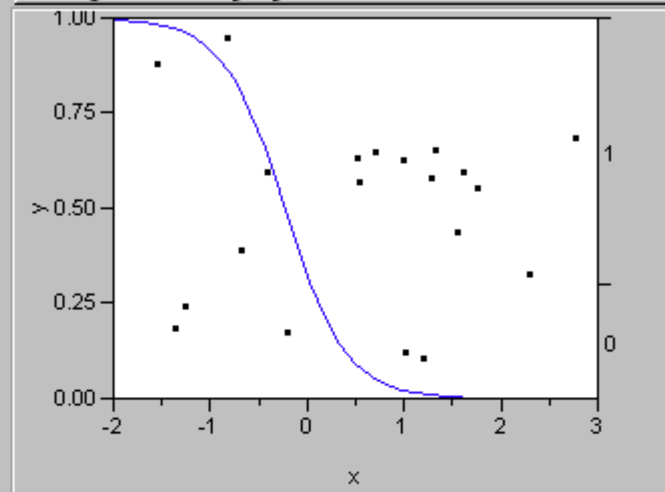
David Madigan

Training Data for a Logistic Regression Model

	x	y
1	-0.8295888	1
2	-0.4187467	0
3	-0.2015895	0
4	-1.3645905	0
5	1.31729882	1
6	1.01640971	1
7	1.27554669	1
8	2.78164437	1
9	1.55595732	1
10	1.20748755	1
11	-0.6737214	0
12	-1.535182	0
13	0.69754466	1
14	0.5412154	1
15	0.98863218	1
16	2.29068842	1
17	-1.2629932	0
18	1.75089817	1
19	0.51903111	1
20	1.61445784	1

Untitled 1- Fit Y by X

Logistic Fit of y By x



Whole Model Test

Model	-LogLikelihood	DF	ChiSquare	Prob>ChiSq
Difference	8.402864	1	16.80573	<.0001
Full	3.814422			
Reduced	12.217286			

RSquare (U)	0.6878
Observations (or Sum Wgts)	20

Converged by Gradient

Parameter Estimates

Term	Estimate	Std Error	ChiSquare	Prob>ChiSq
Intercept	-0.7260123	1.0029956	0.52	0.4692
x	-3.1307611	1.4590883	4.60	0.0319

For log odds of 0/1

	x	y	yhat
21	-1.8826435	0	0.00569376
22	-1.7042119	0	0.00991067
23	-1.3975266	0	0.02547486
24	-1.2538216	0	0.03937468
25	-1.0572248	0	0.07049479
26	-1.0127313	0	0.08018405
27	-0.9385969	0	0.09905148
28	-0.4356167	0	0.34672181
29	-0.2414375	0	0.49357551
30	-0.0555006	0	0.6355921
31	0.04653626	1	0.70592175
32	0.12306672	1	0.75309706
33	0.40439298	0	0.88035014
34	0.58503442	1	0.92831953
35	0.88483088	0	0.97067413
36	1.00772934	1	0.97984995
37	1.0785977	1	0.98379361
38	1.08545156	1	0.98413212
39	1.55540951	1	0.99631
40	2.6417006	1	0.99987642

predicted probabilities

Test Data

Suppose we use a cutoff of 0.5…

predicted outcome	actual outcome: 1	actual outcome: 0
1	8	3
0	0	9

More generally…

actual outcome

		1	0
predicted outcome	1	*a*	*b*
	0	*c*	*d*

misclassification rate: $\frac{b + c}{a+b+c+d}$

sensitivity: $\frac{a}{a+c}$

(aka recall)

specificity: $\frac{d}{b+d}$

predicitive value positive: $\frac{a}{a+b}$

(aka precision)

Suppose we use a cutoff of 0.5…

		actual outcome 1	actual outcome 0
predicted outcome	1	8	3
	0	0	9

sensitivity: $\frac{8}{8+0}$ = 100%

specificity: $\frac{9}{9+3}$ = 75%

Suppose we use a cutoff of 0.8…

actual outcome

		1	0
predicted outcome	1	6	2
	0	2	10

sensitivity: $\frac{6}{6+2}$ = 75%

specificity: $\frac{10}{10+2}$ = 83%

- Note there are 20 possible thresholds

- ROC computes sensitivity and specificity for all possible thresholds and plots them

- Note if threshold = minimum

  c=d=0 so sens=1; spec=0

- If threshold = maximum

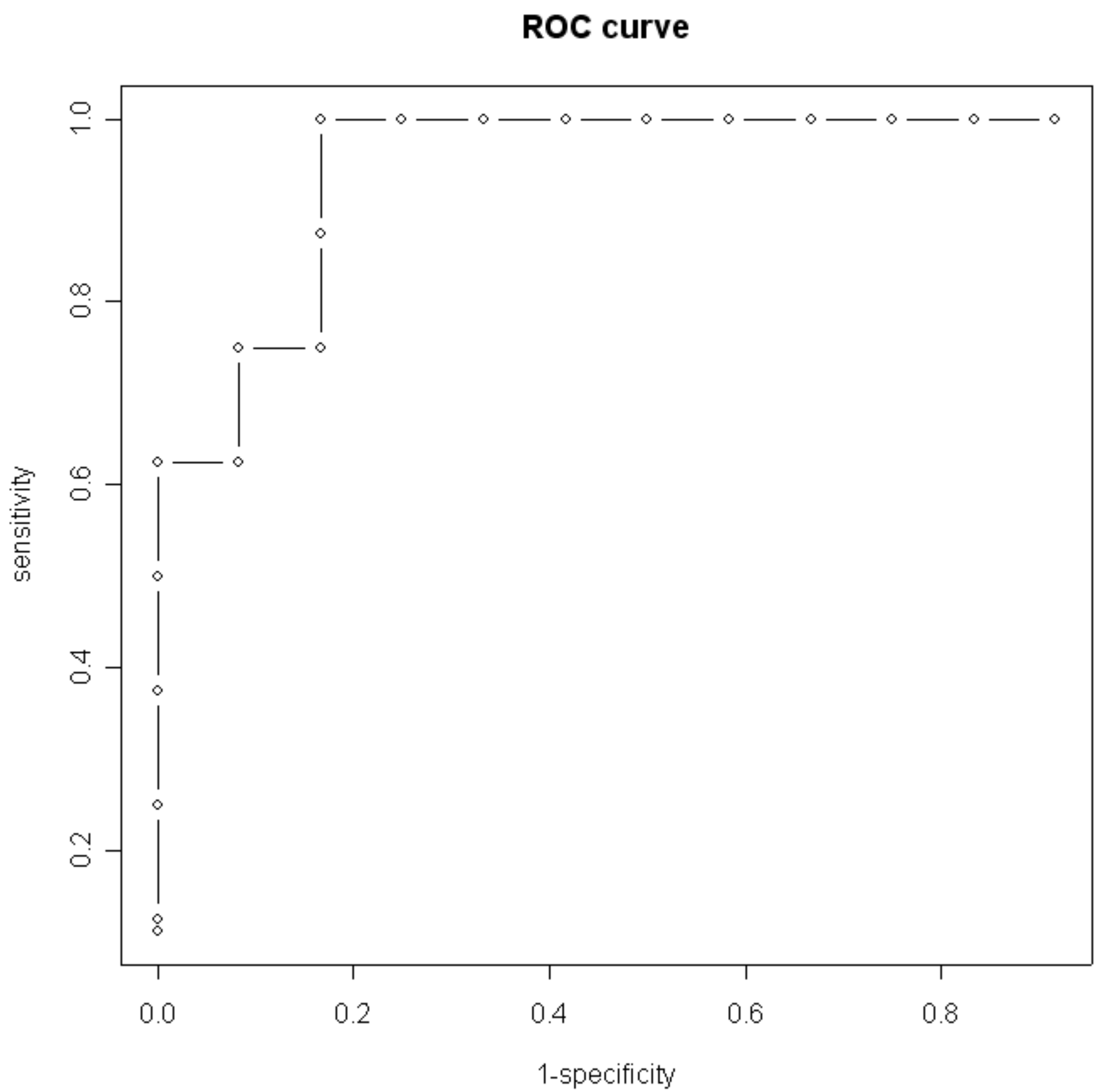
  a=b=0 so sens=0; spec=1

	actual outcome 1	actual outcome 0
1	*a*	*b*
0	*c*	*d*

A1 *fx*

	A	C	D	E	F	G	H	I
1			**a**	**b**	**c**	**d**	**sensitivity**	**specificity**
2	0	0.005694	8	11	0	1	1	0.083333
3	0	0.009911	8	10	0	2	1	0.166667
4	0	0.025475	8	9	0	3	1	0.25
5	0	0.039375	8	8	0	4	1	0.333333
6	0	0.070495	8	7	0	5	1	0.416667
7	0	0.080184	8	6	0	6	1	0.5
8	0	0.099051	8	5	0	7	1	0.583333
9	0	0.346722	8	4	0	8	1	0.666667
10	0	0.493576	8	3	0	9	1	0.75
11	0	0.635592	8	2	0	10	1	0.833333
12	1	0.705922	7	2	1	10	0.875	0.833333
13	1	0.753097	6	2	2	10	0.75	0.833333
14	0	0.88035	6	1	2	11	0.75	0.916667
15	1	0.92832	5	1	3	11	0.625	0.916667
16	0	0.970674	5	0	3	12	0.625	1
17	1	0.97985	4	0	4	12	0.5	1
18	1	0.983794	3	0	5	12	0.375	1
19	1	0.984132	2	0	6	12	0.25	1
20	1	0.99631	1	0	7	12	0.125	1
21	1	0.999876	1	0	8	12	0.111111	1
22								
23								

```
sens<-c(1,1,1,1,1,1,1,1,1,1,0.875,0.75,0.75,0.625,0.625,0.5,0.375,0.25,0.125,0.11111
spec<-c(0.083333333,0.166666667,0.25,0.333333333,0.416666667,0.5,0.583333333,0.66666
33333,0.916666667,0.916666667,1,1,1,1,1,1)
plot(1-spec,sens,type="b",xlab="1-specificity",ylab="sensitivity",main="ROC curve")
```



ROC curve
sensitivity
1-specificity
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8

- “Area under the curve” is a common measure of predictive performance
- R library “verification” has roc.area and roc.plot:

```
roc.area(c(1,0,1,0,1,0),c(0.9,0.6,0.4,0.7,0.8,0.1))
$A
[1] 0.7777778
$n.total
[1] 6
$n.events
[1] 3
$n.noevents
[1] 3
$p.value
[1] 0.2
```

- Squared error also used: $\Sigma(y_i - y\text{hat})^2$

  also known as the “Brier Score”