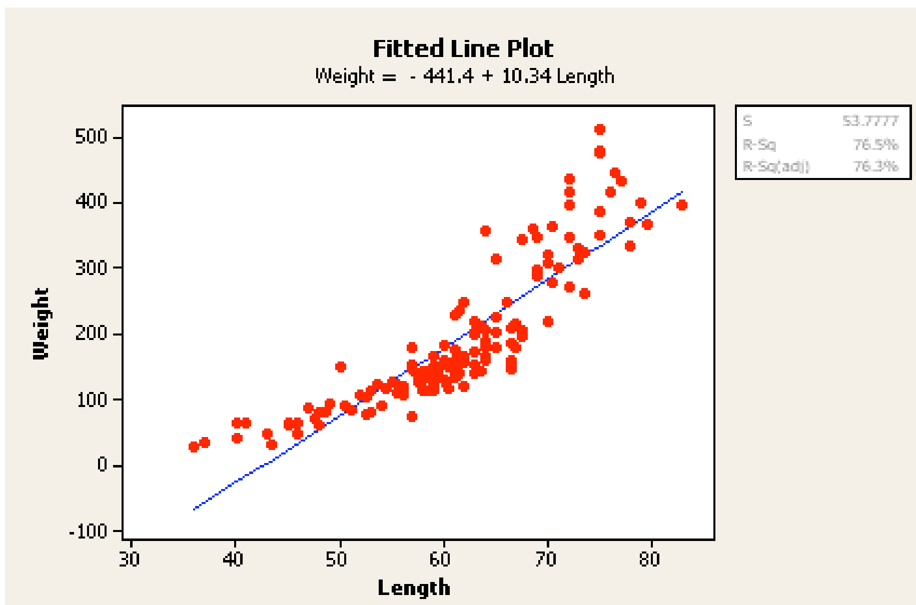
Logistic & Tobit Regression

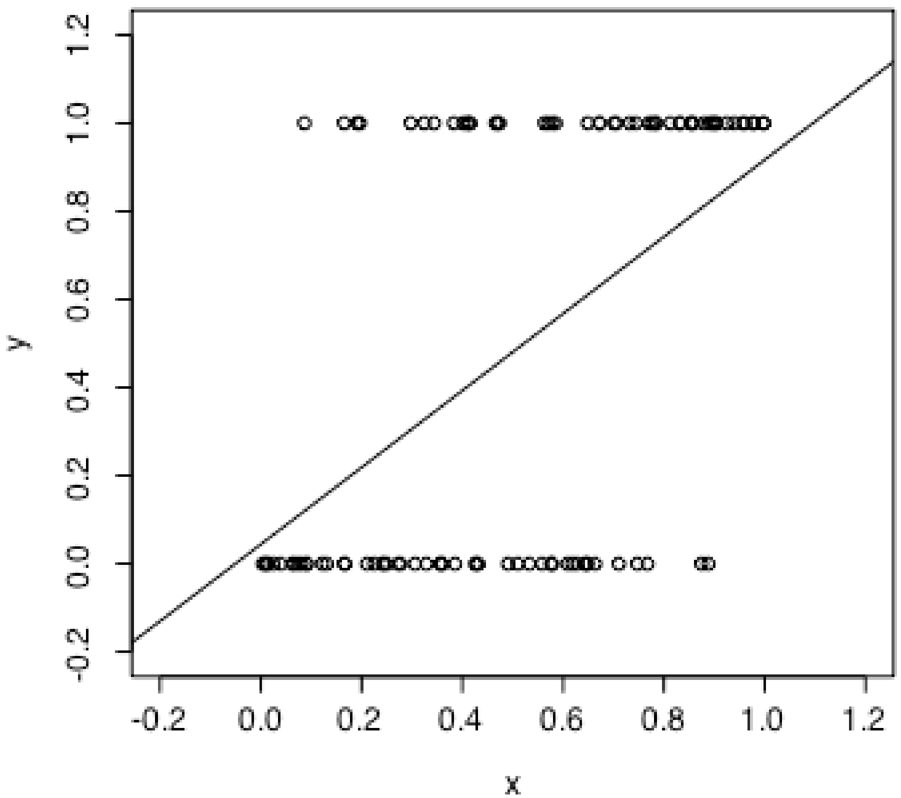
Different Types of Regression

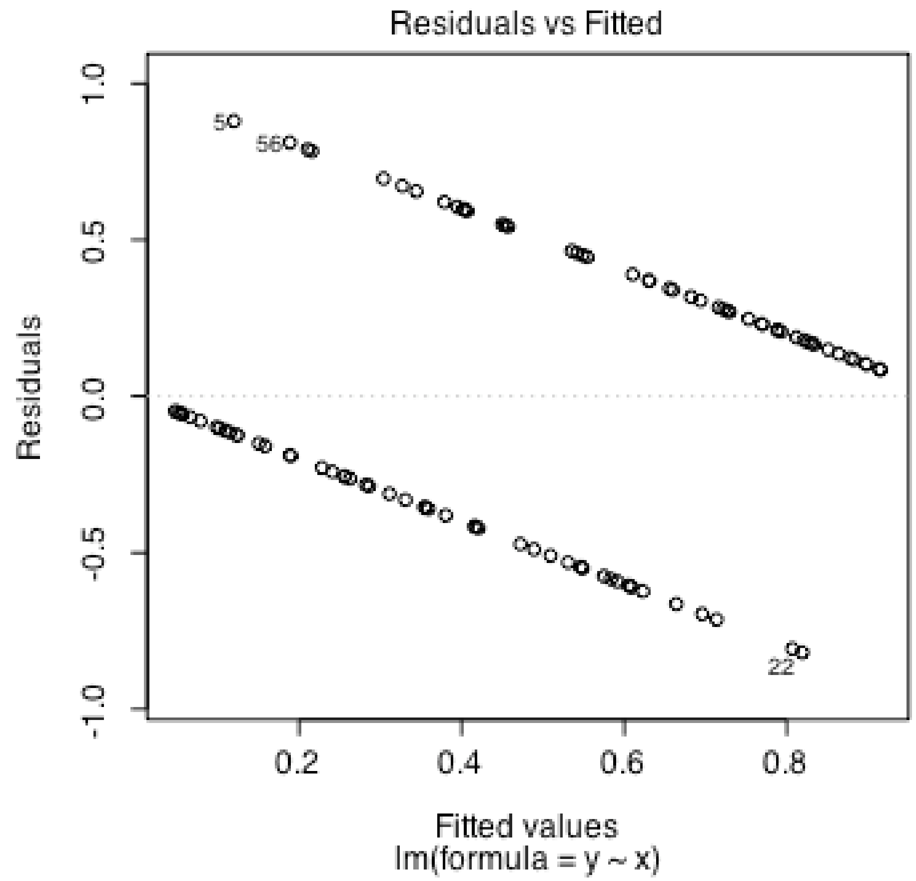
- Means → Linear regression
- Odds → Logistic regression
- Rates → Poisson regression
- Hazards → Proportional Hazards regression
- Quantiles → Parametric survival regression



Binary Regression

- Imagine a simple linear regression setting with a continuous predictor of interest and a binary response of interest (D)
- What would a scatterplot of the data look like?
- How would a linear regression line fit these data?


y
x
-0.2
0.0
0.2
0.4
0.6
0.8
1.0
1.2


Residuals vs Fitted
Residuals
Fitted values
lm(formula = y ~ x)
1.0
0.5
0.0
-0.5
-1.0
0.2
0.4
0.6
0.8
5
56
22

- If our response D is binary, we are generally interested in inference about P(D).
  - If our predictor X is related to D, then we want to know P(D|X)
- Our software is happy to perform a linear regression with a 0/1 response and a continuous predictor.
- So what's the problem?

•Proportions/Probabilities have to be between 0 and 1

–With binary data, the variance within a group depends on the mean

Regression with Binary Response

- Instead of using linear regression to model probabilities, we use logistic regression to model the log odds.
- The odds of an event are between 0 and infinity

$$\text{odds} = \text{prob} / (1 - \text{prob})$$

- log(odds) are between negative infinity and positive infinity (even better)

Simple Logistic Regression

- Modeling odds of binary response *Y* on predictor *X*

Distribution $\Pr(D_i = 1 \mid X_i) = p_i$

Model $\text{logit}(p_i) = \log\left(\dfrac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 \times X_i$

$X_i = 0$ $\quad$ $\log \text{odds} = \beta_0$

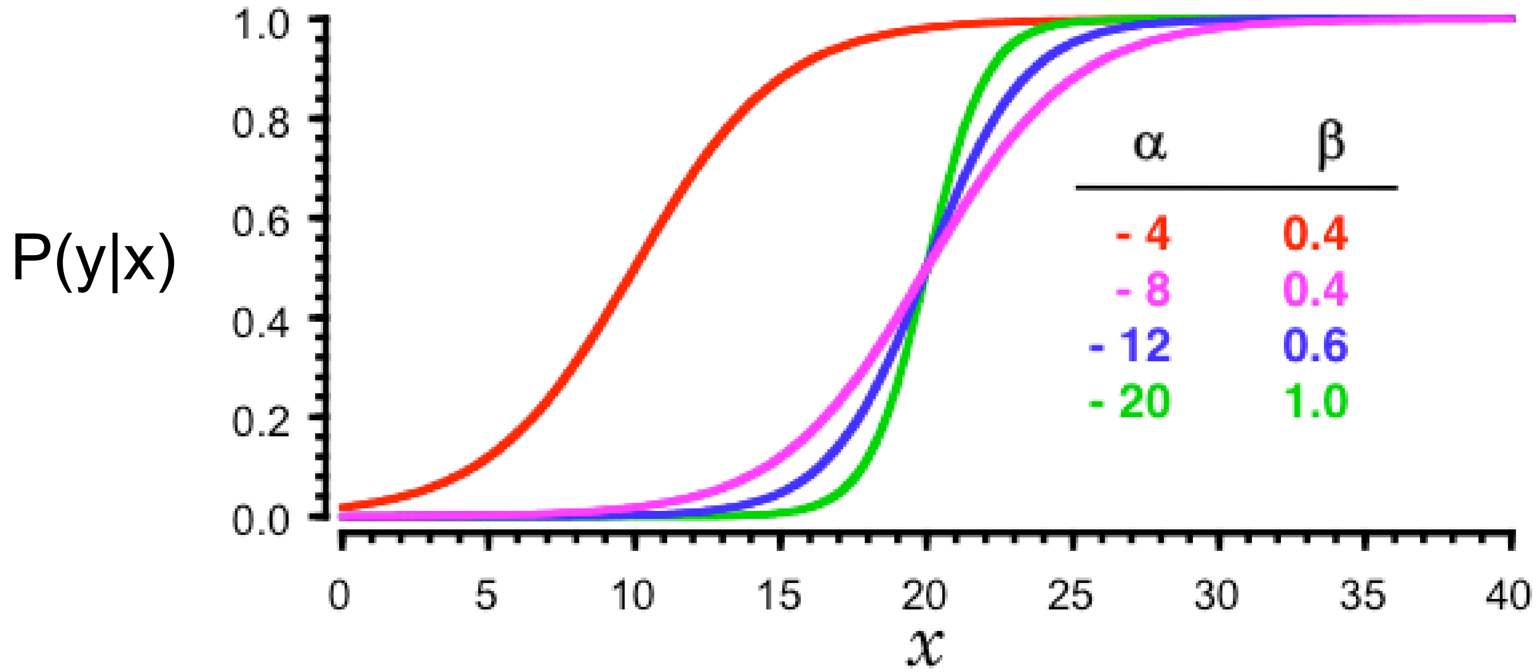
$X_i = x$ $\quad$ $\log \text{odds} = \beta_0 + \beta_1 \times x$

$X_i = x+1$ $\quad$ $\log \text{odds} = \beta_0 + \beta_1 \times x + \beta_1$

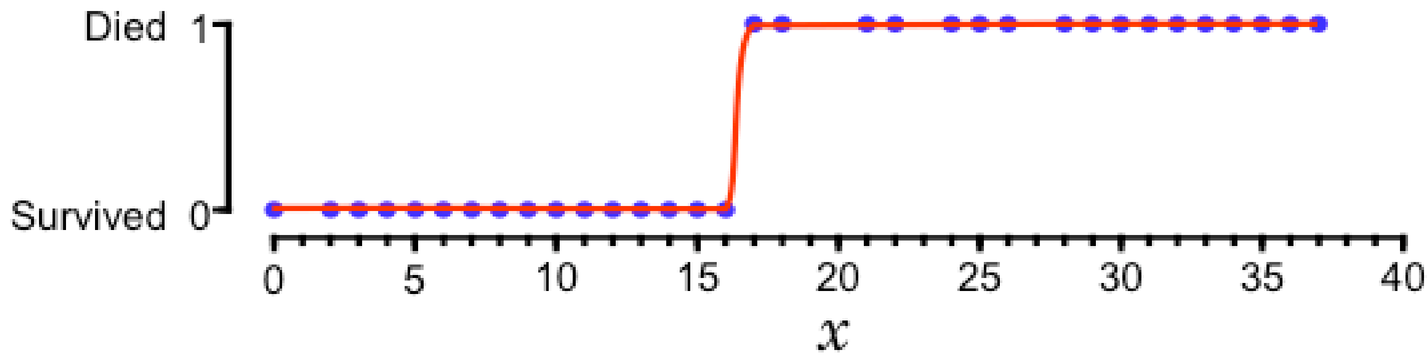
Logistic transformation

$$P(y|x) = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}}$$

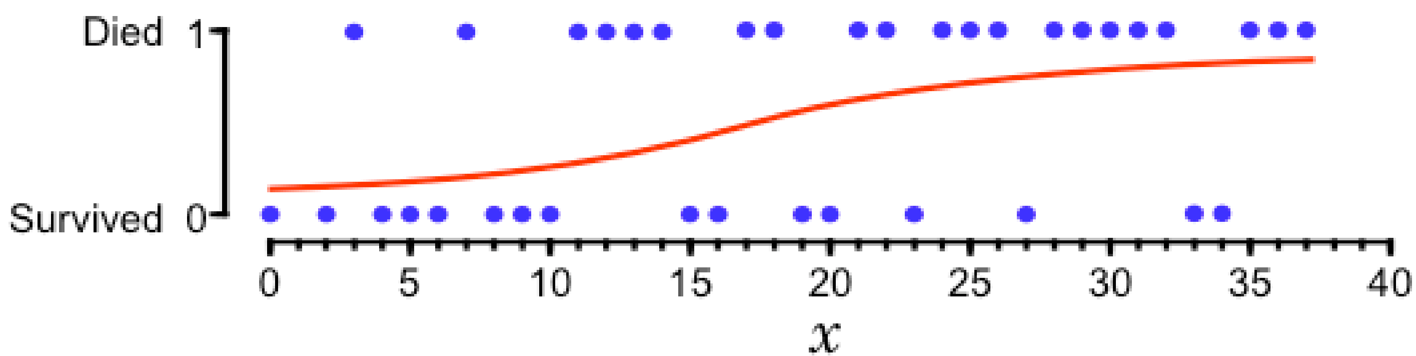
$$\underbrace{\ln\left[\frac{P(y|x)}{1-P(y|x)}\right]}_{\text{logit of } P(y|x)} = \alpha + \beta x$$


P(y|x)
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
30
35
40
x
α β
- 4 0.4
- 8 0.4
- 12 0.6
- 20 1.0

Data that has a sharp survival cut off point between patients who live or die should have a large value of β.



Data with a lengthy transition from survival to death should have a low value of β.



Log likelihood

$$l(w) = \sum_{i=1}^{N} y_i \log p(x_i; w) + (1 - y_i) \log(1 - p(x_i; w))$$

Log likelihood

$$l(w) = \sum_{i=1}^{N} y_i \log p(x_i;w) + (1 - y_i)\log(1 - p(x_i;w))$$

$$= \sum_{i=1}^{N} y_i \log \frac{p(x_i;w)}{(1 - p(x_i;w))} + \log(\frac{1}{1+e^{x_i w}})$$

$$= \sum_{i=1}^{N} y_i x_i w - \log(1 + e^{x_i w})$$

- Note: this likelihood is a concave

Maximum likelihood estimation

$$\frac{\partial}{\partial w_j} l(w) = \frac{\partial}{\partial w_j} \sum_{i=1}^{N} \{y_i x_i w - \log(1 + e^{x_i w})\}$$

$$\ldots^{N}_{=1} x_{ij} (\underbrace{y_i - p(x_i, w)}_{\text{prediction error}})$$

No closed form solution!

Common (but not only) approaches:

Numerical Solutions:

- Line Search
- Simulated Annealing
- Gradient Descent
- Newton's Method

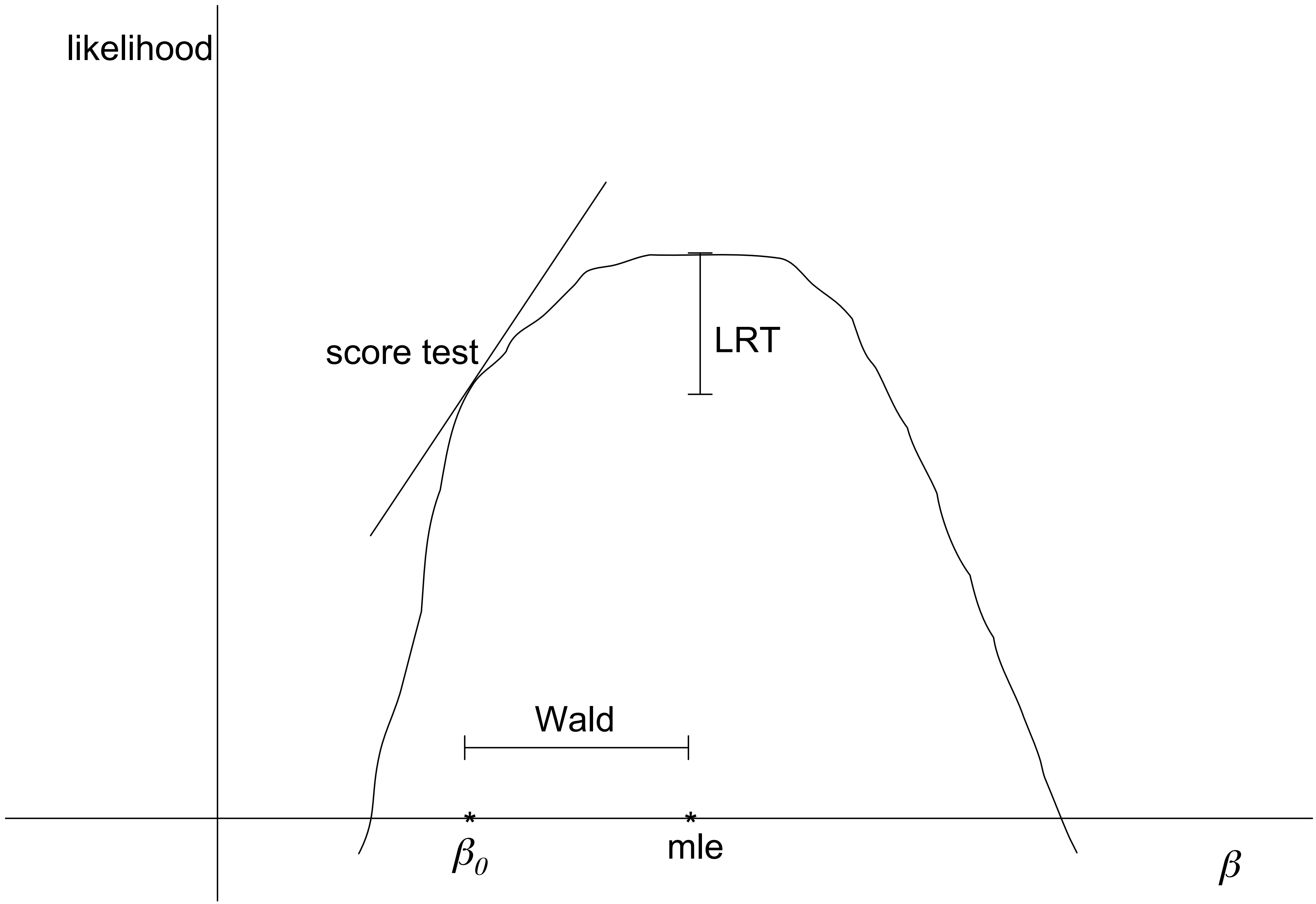
Statistical Inference in Logistic Regression

Inferential Methods in Logistic Regression

- Unlike linear regression, in logistic regression the Wald, score, and likelihood ratio tests are only asymptotically equivalent
  - The Wald test can be poorly behaved in small samples
    - Thus it is important to know how to perform a LRT for datasets of smaller size

Likelihood Ratio Tests

- A likelihood ratio test compares the fit of the full model relative to that of the restricted model
  - Key points
    - The two models <u>must</u> be hierarchical: The full model must contain all terms present in the restricted model
      - E.g., compare model of age and height to a model just of age
      - (Cannot compare model of age to a model of height)
    - The same cases should be used to fit each model
      - Watch out for missing data!!!!


likelihood
score test
LRT
Wald
β_0
mle
β

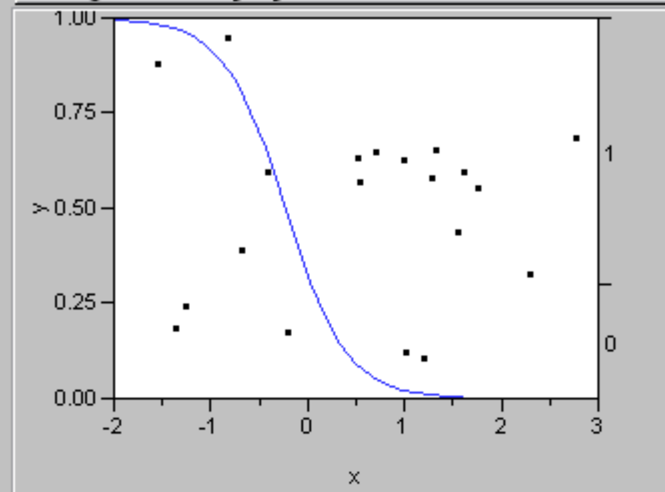
Measuring the Performance of a Binary Classifier

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…

actual outcome

predicted outcome	1	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	actual outcome 0
predicted outcome	1	6	2
	0	2	10

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

specificity: $\dfrac{10}{10+2}$ = 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

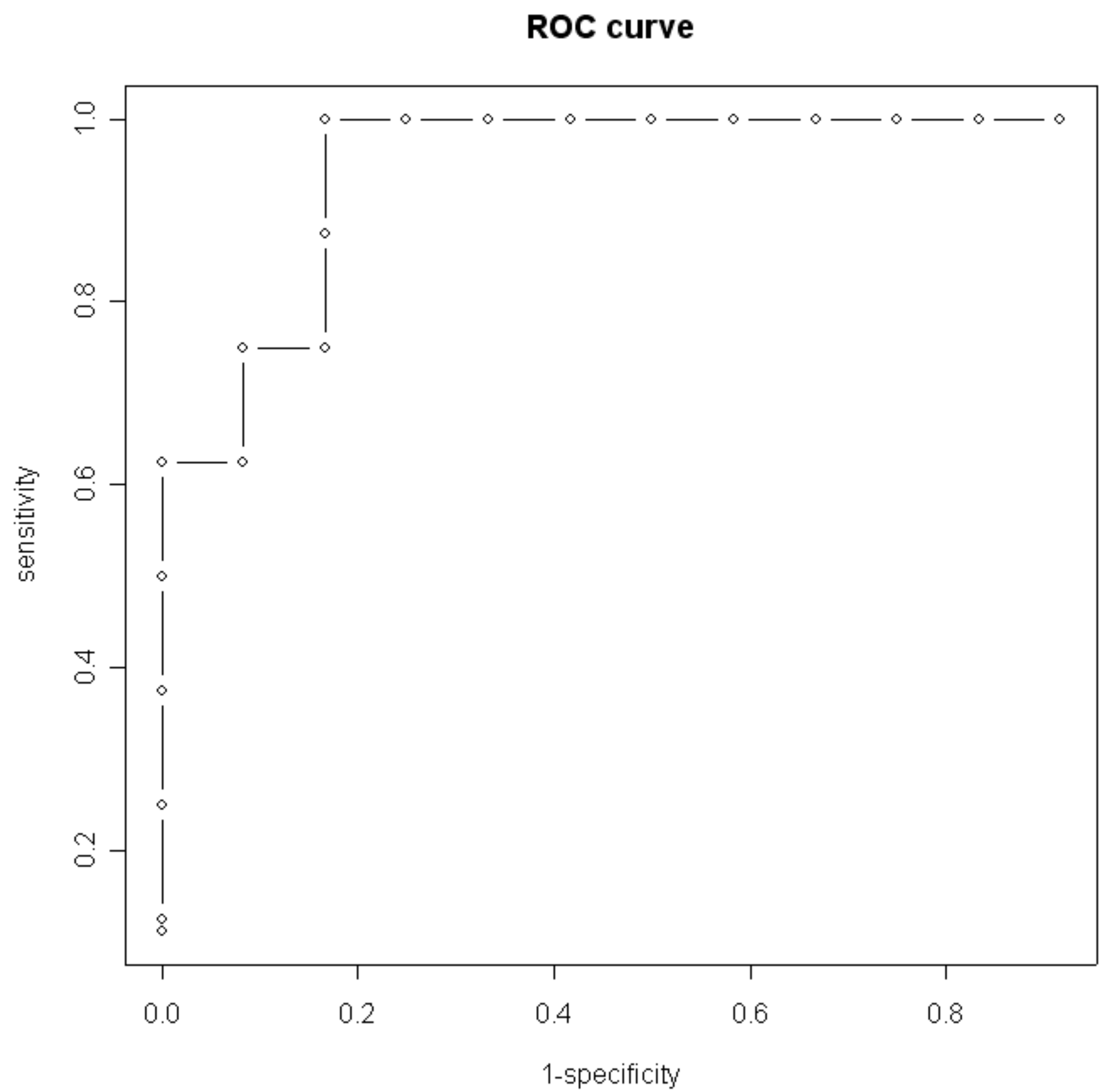
  a=*b*=0 so sens=0; spec=1

	actual outcome	
	1	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

- So is squared error: $\Sigma(y_i\text{-}y\text{hat})^2$

  also known as the “Brier Score”

Penalized Logistic Regression

Ridge Logistic Regression

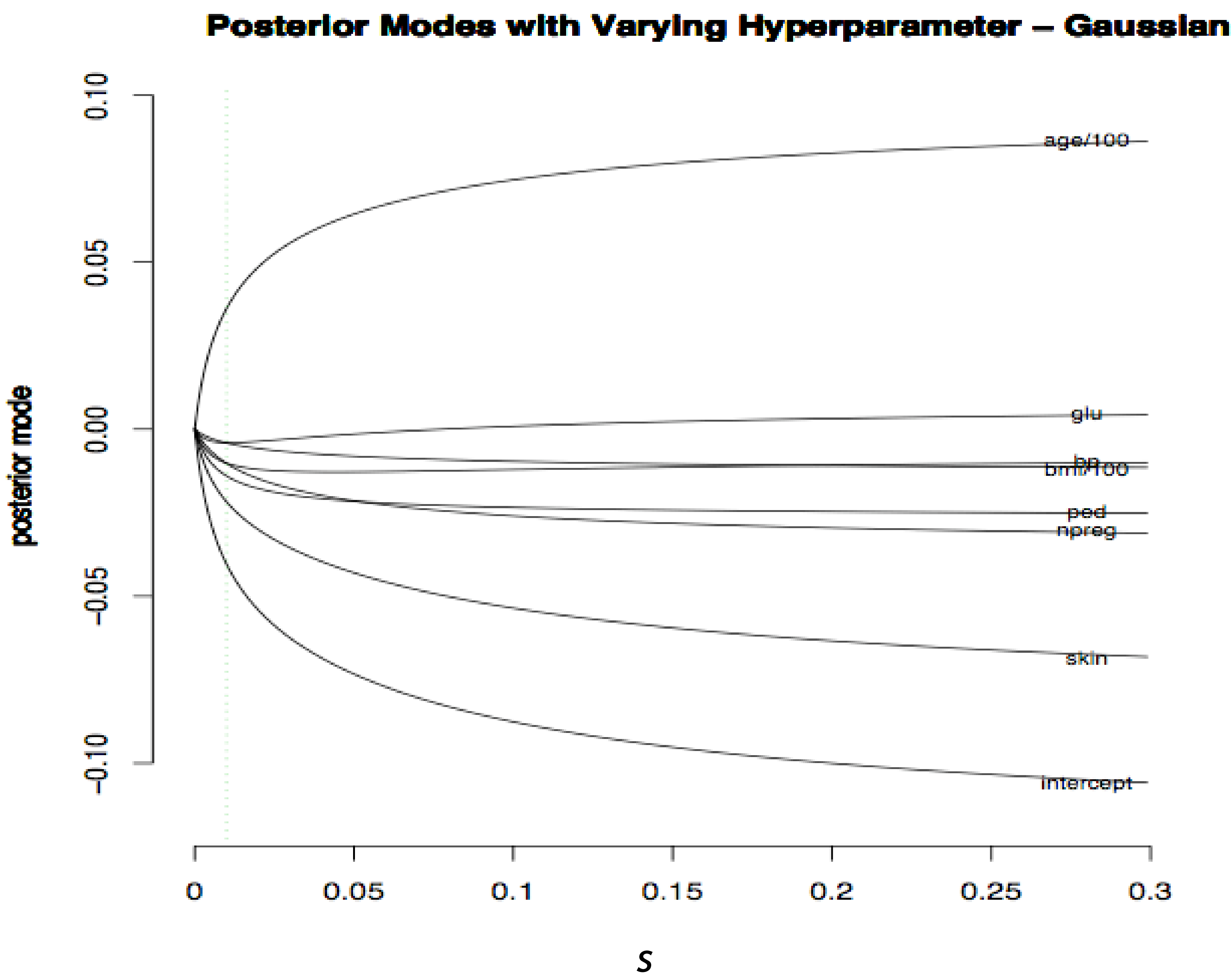
Maximum likelihood plus a constraint:

$$\sum_{j=1}^{p} \beta_j^2 \leq s$$

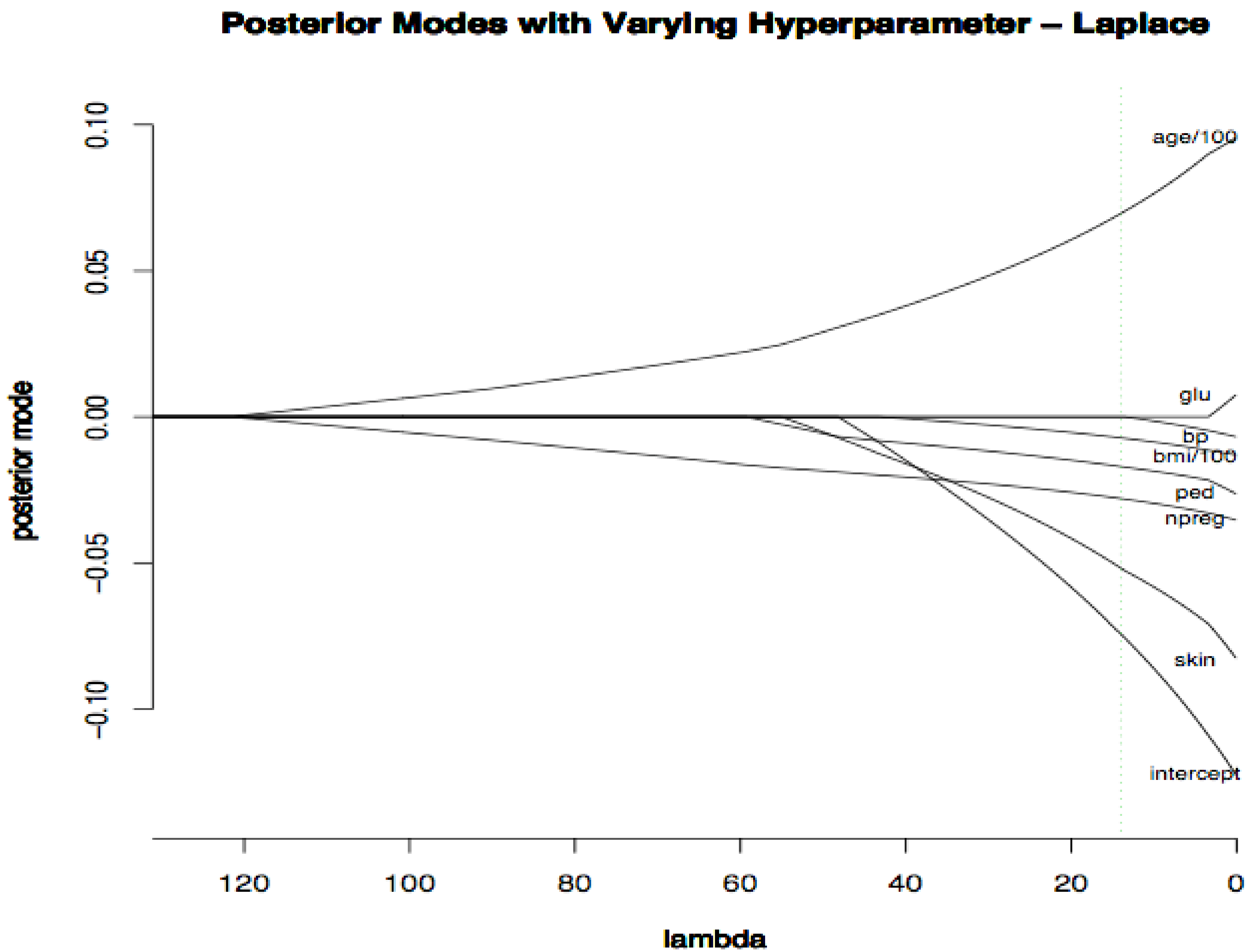
Lasso Logistic Regression

Maximum likelihood plus a constraint:

$$\sum_{j=1}^{p} \left|\beta_j\right| \leq s$$



Posterior Modes with Varying Hyperparameter – Gaussian
posterior mode
0.10
0.05
0.00
−0.05
−0.10
age/100
glu
bp
bmi/100
ped
npreg
skin
intercept
0
0.05
0.1
0.15
0.2
0.25
0.3
S


Posterior Modes with Varying Hyperparameter – Laplace
posterior mode
0.10
0.05
0.00
−0.05
−0.10
120
100
80
60
40
20
0
lambda
age/100
glu
bp
bmi/100
ped
npreg
skin
intercept

http://www.bayesianregression.org

http://www.stanford.edu/~boyd/l1_logreg/

Polytomous Logistic Regression (PLR)

$$P(y_i = k \mid \mathbf{x_i}) = \frac{\exp(\vec{\beta}_k \mathbf{x}_i)}{\sum_{k'} \exp(\vec{\beta}_{k'} \mathbf{x}_i)}$$

- Elegant approach to multiclass problems
- Also known as *polychotomous LR*, *multinomial LR*, and, ambiguously, *multiple LR* and *multivariate LR*

1-of-K Sample Results: brittany-l

Feature Set	% errors	Number of Features
“Argamon” function words, raw tf	74.8	380
POS	75.1	44
1suff	64.2	121
1suff*POS	50.9	554
2suff	40.6	1849
2suff*POS	34.9	3655
3suff	28.7	8676
3suff*POS	27.9	12976
3suff+POS+3suff*POS+Argamon	27.6	22057
All words	23.9	52492

4.6 million parameters

89 authors with at least 50 postings. 10,076 training documents, 3,322 test documents.

BMR-Laplace classification, default hyperparameter

Generalized Linear Model

1. Outcome $y = (y_1, \ldots, y_n)$.
2. Linear predictor $X\beta$ with X is $n \times k$ predictor matrix and β is $k \times 1$ vector of coefficients.
3. Link function g to transform $X\beta$ into $\hat{y} = g^{-1}(X\beta)$.
4. Data distribution$p(y|\hat{y})$.
5. Other parameters: variances, overdispersions, cutpoints, etc

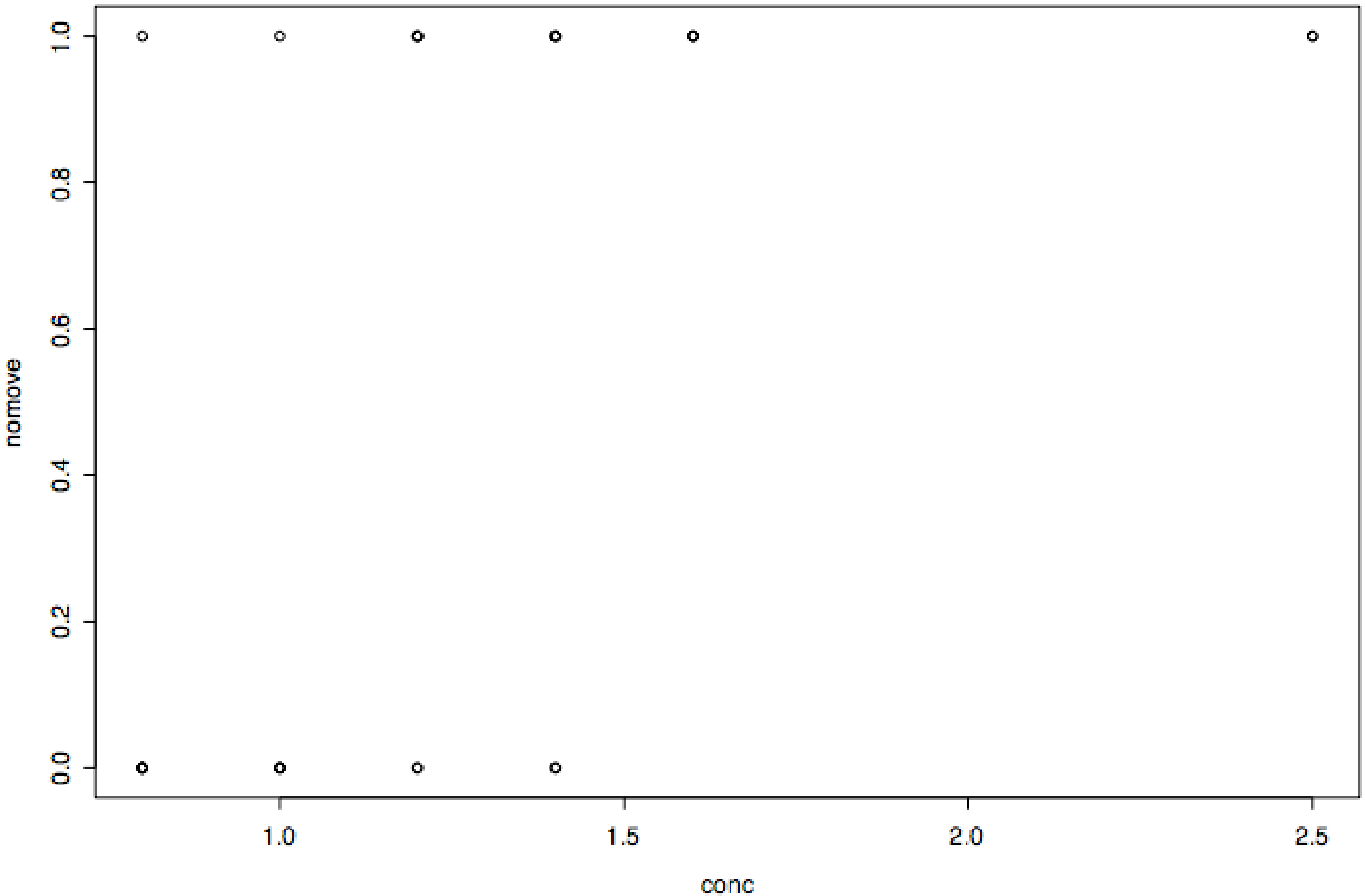
So far we studied:

1. *Linear regression*: $g(u) = u$ and $y \sim N(X\beta, \sigma^2)$.
2. *Logistic regression*: $g(u) = \text{logit}(u)$ and $P(y = 1) = \hat{y}$.

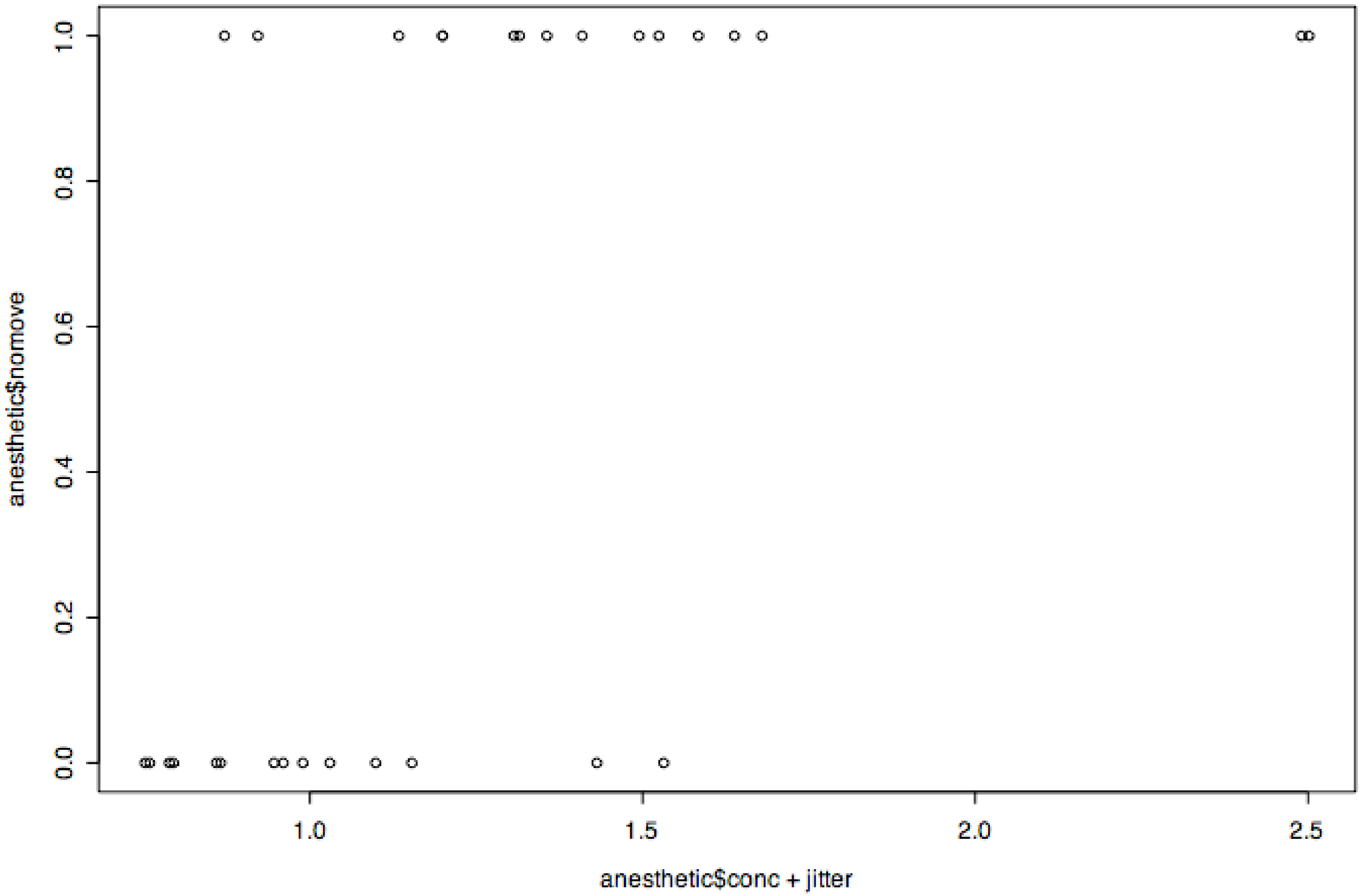
Logistic Regression in R

```
> anesthetic
  move conc   logconc nomove
1    0  1.0 0.0000000      1
2    1  1.2 0.1823216      0
3    0  1.4 0.3364722      1
4    1  1.4 0.3364722      0
5    1  1.2 0.1823216      0
6    0  2.5 0.9162907      1
```

```
plot(nomove~conc,data=anesthetic)
```



1.0
0.8
0.6
0.4
0.2
0.0
nomove
1.0
1.5
2.0
2.5
conc


anesthetic$nomove
0.0
0.2
0.4
0.6
0.8
1.0
1.0
1.5
2.0
2.5
anesthetic$conc + jitter

```
> anes.logit <- glm(nomove ~ conc, family=binomial(link="logit"),
data=anesthetic)
> summary(anes.logit)

Call:
glm(formula = nomove ~ conc, family = binomial(link = "logit"),
    data = anesthetic)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.76666  -0.74407   0.03413   0.68666   2.06900

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   -6.469      2.418  -2.675  0.00748 **
conc           5.567      2.044   2.724  0.00645 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 41.455  on 29  degrees of freedom
Residual deviance: 27.754  on 28  degrees of freedom
AIC: 31.754

Number of Fisher Scoring iterations: 5
```

Deviance (statistics)

From Wikipedia, the free encyclopedia

In statistics, **deviance** is a quantity whose expected values can be used for statistical hypothesis testing.

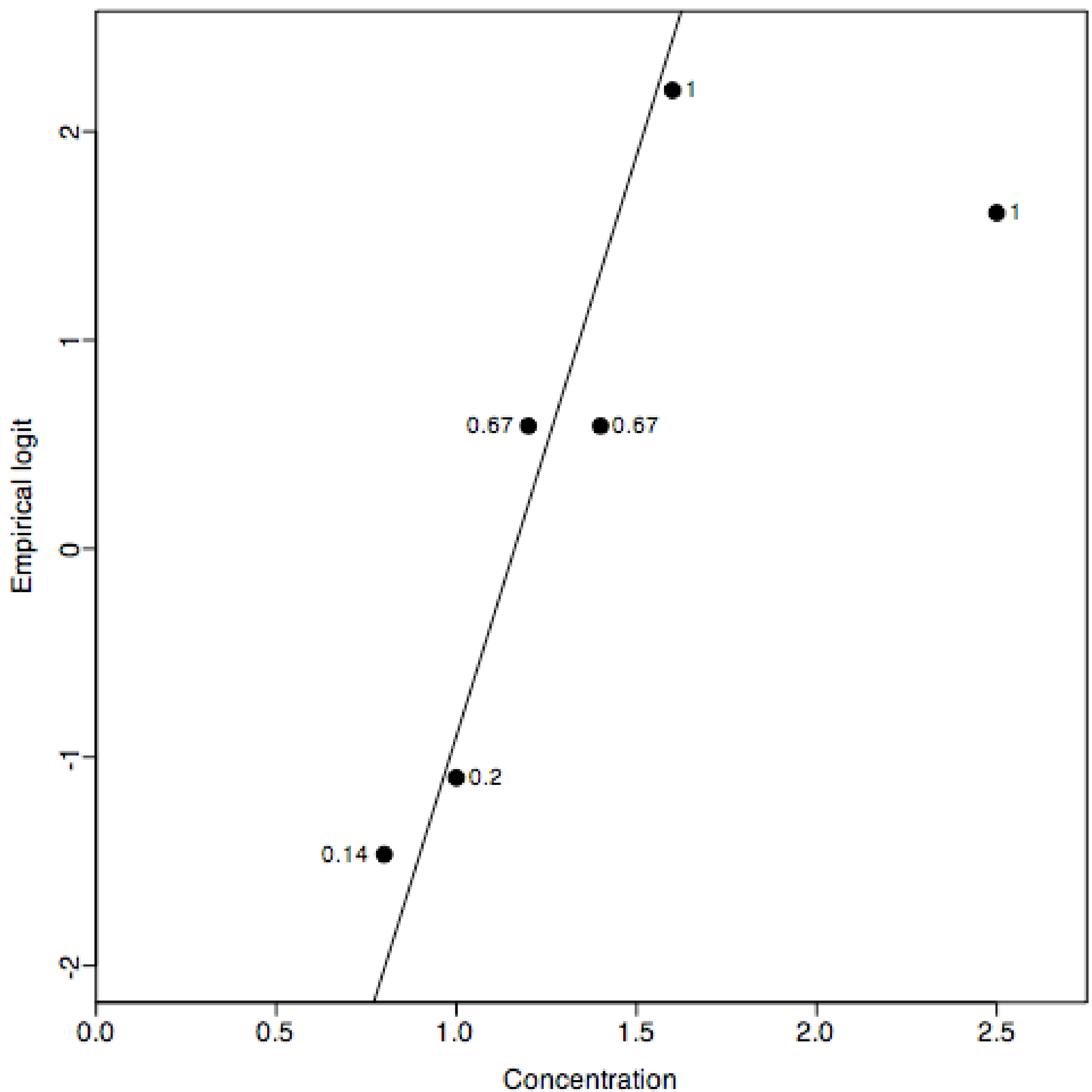
It is defined as

$$D(y,\theta) = -2\log[p(y|\theta)].$$

As a function of θ with *y* treated as fixed, it is -2 times the log-likelihood. In the framework of the generalized linear model, if θ is the true parameter, $D(y, \theta)$ follows a chi-squared distribution.

Deviance Residuals:

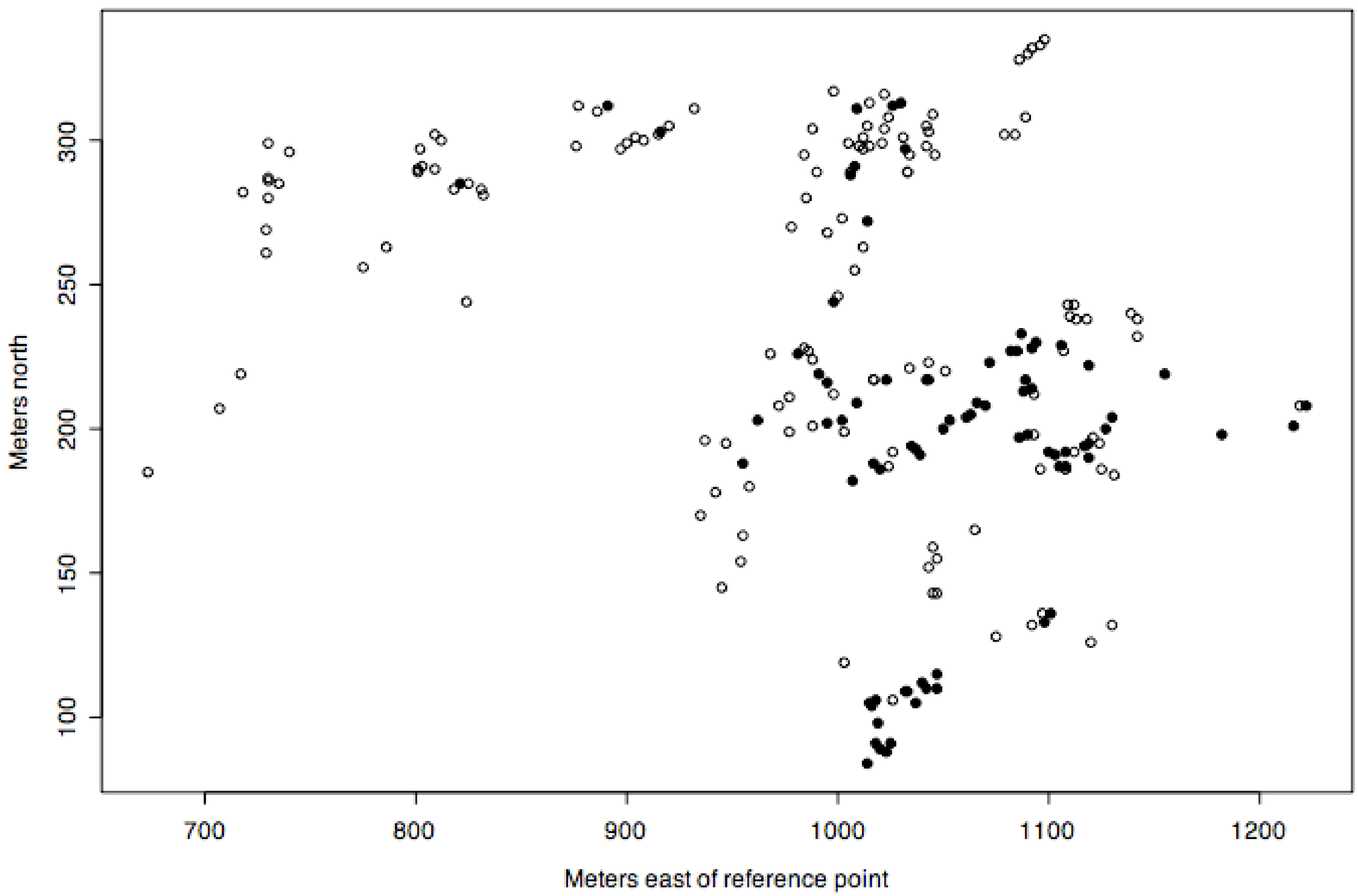
$$r_D(i) = sign(y_i - \hat{y}_i)\sqrt{D(y_i,\theta)}$$



Empirical logit
Concentration
0.0
0.5
1.0
1.5
2.0
2.5
-2
-1
0
1
2
1
1
0.67
0.67
0.2
0.14

```
> frogs
  pres.abs northing easting altitude distance NoOfPools NoOfSites   avrain  meanmin  meanmax
2        1      115    1047     1500      500       232         3 155.0000 3.566667 14.00000
3        1      110    1042     1520      250        66         5 157.6667 3.466667 13.80000
4        1      112    1040     1540      250        32         5 159.6667 3.400000 13.60000
5        1      109    1033     1590      250         9         5 165.0000 3.200000 13.16667
6        1      109    1032     1590      250        67         5 165.0000 3.200000 13.16667
7        1      106    1018     1600      500        12         4 167.3333 3.133333 13.06667
8        1      105    1015     1600      250        30         3 167.3333 3.100000 13.06667
```

```
data(frogs)
plot(northing ~ easting, data=frogs, pch=c(1,16)[frogs$pres.abs+1],
     xlab="Meters east of reference point", ylab="Meters north")
```



300
250
200
150
100
Meters north
700
800
900
1000
1100
1200
Meters east of reference point

```
> frogs.glm0 <- glm(pres.abs ~ .,data=frogs,family=binomial(link=logit))
> summary(frogs.glm0)

Call:
glm(formula = pres.abs ~ ., family = binomial(link = logit),
    data = frogs)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8987  -0.7987  -0.2735   0.8035   2.6991

Coefficients:
              Estimate Std. Error   z value Pr(>|z|)
(Intercept) -1.635e+02  2.153e+02    -0.759  0.44764
northing     1.041e-02  1.654e-02     0.630  0.52901
easting     -2.158e-02  1.268e-02    -1.702  0.08872 .
altitude     7.091e-02  7.705e-02     0.920  0.35745
distance    -4.835e-04  2.060e-04    -2.347  0.01893 *
NoOfPools    2.968e-02  9.444e-03     3.143  0.00167 **
NoOfSites    4.294e-02  1.095e-01     0.392  0.69482
avrain      -4.058e-05  1.300e-01 -0.000312  0.99975
meanmin      1.564e+01  6.479e+00     2.415  0.01574 *
meanmax      1.708e+00  6.809e+00     0.251  0.80198

  Null deviance: 279.99  on 211  degrees of freedom
Residual deviance: 195.66  on 202  degrees of freedom
AIC: 215.66
```

```
> frogs.glm1 <- glm(pres.abs ~ easting + distance + NoOfPools +
meanmin,data=frogs,family=binomial(link=logit))
> summary(frogs.glm1)

Call:
glm(formula = pres.abs ~ easting + distance + NoOfPools + meanmin,
    family = binomial(link = logit), data = frogs)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8082  -0.7942  -0.4048   0.8751   2.9700

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -6.4501010  2.5830600  -2.497 0.012522 *
easting      0.0024885  0.0031229   0.797 0.425533
distance    -0.0005908  0.0001744  -3.387 0.000706 ***
NoOfPools    0.0244347  0.0080995   3.017 0.002555 **
meanmin      1.1130796  0.4191963   2.655 0.007924 **

    Null deviance: 279.99  on 211  degrees of freedom
    Residual deviance: 215.45  on 207  degrees of freedom
    AIC: 225.45
```

AIC not as good…

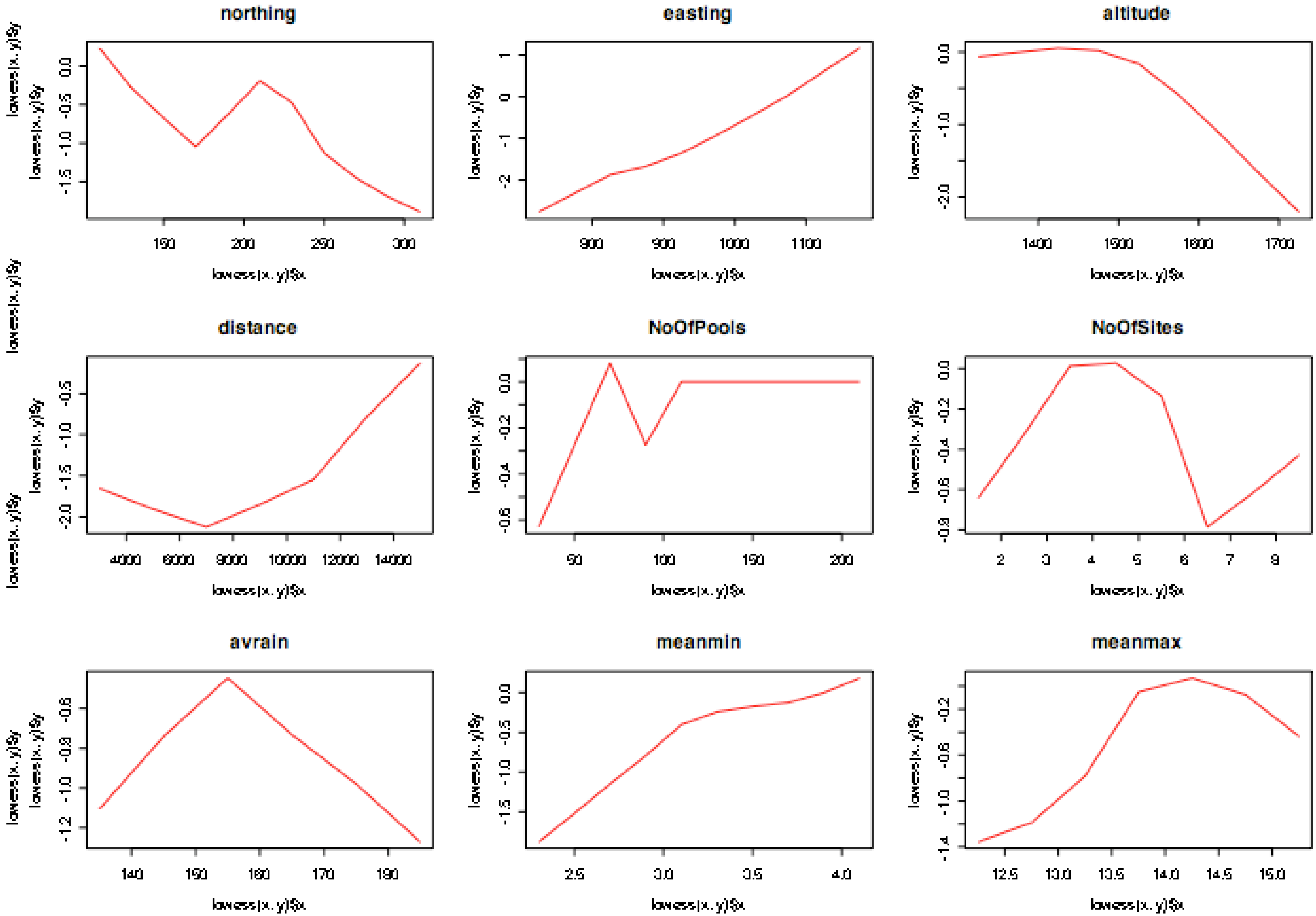
```
> CVbinary(frogs.glm0)

Fold:  5 4 9 6 3 2 1 10 8 7
Internal estimate of accuracy = 0.792
Cross-validation estimate of accuracy = 0.778
> CVbinary(frogs.glm1)

Fold:  4 6 1 10 8 2 3 7 5 9
Internal estimate of accuracy = 0.759
Cross-validation estimate of accuracy = 0.731
```

- CV accuracy estimates quite variable with small datasets
- Best to repeat comparing models on same split

```
all.acc <- numeric(10)
red.acc <- numeric(10)
for (j in 1:10) {
  randsam <- sample (1:10,dim(frogs)[1], replace=TRUE)
  all.acc[j] <- CVbinary(frogs.glm0, rand=randsam)$acc.cv
  red.acc[j] <- CVbinary(frogs.glm1, rand=randsam)$acc.cv
}
```

```
> all.acc
 [1] 0.7688679 0.7783019 0.7783019 0.7830189 0.7735849 0.7735849 0.7735849 0.773584
> red.acc
 [1] 0.7264151 0.7500000 0.7264151 0.7358491 0.7358491 0.7358491 0.7547170 0.745283
```

```
par(mfrow=c(3,3))
for (i in 2:10) {
  ints <- pretty(frogs[,i],n=10)
  J <- length(ints)-2
  y <- n <- x <- rep(0,J-1)
  for (j in 2:J) {
        temp <- frogs[((frogs[,i]>ints[j]) & (frogs[,i]<=ints[j+1])),];
        y[j-1] <- sum(temp$pres.abs);
        n[j-1] <- dim(temp)[1];
         x[j-1] <- (ints[j]+ints[j+1])/2;
  }
  y <- logit((y+0.5)/(n+1))
  # plot(x,y,type="b",col="red")
  # par(new=TRUE)
  plot(lowess(x,y), main=names(frogs)[i], type="l", col="red")
}
```


northing
easting
altitude
distance
NoOfPools
NoOfSites
avrain
meanmin
meanmax
lowess(x, y)$x
lowess(x, y)$y

```
> frogs.glm2 <- glm(pres.abs ~ northing + easting +
altitude + distance + I(distance^2)+ NoOfPools + NoOfSites
+ avrain + meanmin + meanmax,
data=frogs,family=binomial(link=logit))
```

```
 Estimate Std. Error z value Pr(>|z|)
(Intercept)   -2.006e+02  2.176e+02  -0.922  0.35658
northing       4.852e-03  1.656e-02   0.293  0.76952
easting       -2.246e-02  1.271e-02  -1.767  0.07717 .
altitude       8.864e-02  7.786e-02   1.138  0.25491
distance      -7.705e-04  2.815e-04  -2.738  0.00619 **
I(distance^2)  4.270e-08  2.385e-08   1.790  0.07342 .
NoOfPools      2.992e-02  9.516e-03   3.144  0.00167 **
NoOfSites      7.765e-04  1.120e-01   0.007  0.99447
avrain        -4.526e-02  1.291e-01  -0.350  0.72598
meanmin        1.618e+01  6.532e+00   2.477  0.01325 *
meanmax        2.966e+00  6.873e+00   0.432  0.66610
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 279.99  on 211  degrees of freedom
Residual deviance: 193.57  on 201  degrees of freedom
AIC: 215.57
```

```
glm(formula = pres.abs ~ northing + easting + altitude + log(distance) +
    NoOfPools + NoOfSites + avrain + meanmin + meanmax, family =
binomial(link = logit),
    data = frogs)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0974  -0.7644  -0.2583   0.7443   2.6454

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)   -1.435e+02  2.130e+02  -0.674  0.50061
northing       9.661e-03  1.576e-02   0.613  0.53992
easting       -1.874e-02  1.280e-02  -1.464  0.14322
altitude       6.550e-02  7.579e-02   0.864  0.38749
log(distance) -7.999e-01  2.551e-01  -3.136  0.00171 **
NoOfPools      2.813e-02  9.499e-03   2.961  0.00306 **
NoOfSites      1.897e-03  1.104e-01   0.017  0.98629
avrain        -7.682e-03  1.229e-01  -0.062  0.95018
meanmin        1.463e+01  6.531e+00   2.239  0.02513 *
meanmax        1.336e+00  6.690e+00   0.200  0.84174
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 279.99  on 211  degrees of freedom
Residual deviance: 192.47  on 202  degrees of freedom
AIC: 212.47
```

CVbinary: 0.77 - 0.79

Other things to try…

- changepoints
- more log, ^2, exp, etc.

```
> frogs.glm3 <- glm(pres.abs ~ northing + easting + altitude+ log(distance)+
NoOfPools + NoOfSites + avrain + log(meanmin) + meanmax +
I(avrain*(avrain>155)),data=frogs,family=binomial(link=logit))
```

Latent Variable Interpretations

- Suppose our binary dependent variable depends on an unobserved utility index, Y^*
- If Y is discrete—taking on the values 0 or 1 if someone buys a car, for instance
  - Can imagine a continuous variable Y^* that reflects a person's desire to buy the car
    - Y^* would vary continuously with some explanatory variable like income

Logit and Probit Models

- Written formally as

$$Y_i^* = \beta_0 + \beta_1 X_{1i} + \varepsilon_i$$

- If the utility index is "high enough," a person will buy a car

$$Y_i = 1 \text{ if } Y_i^* \geq 0$$

- If the utility index is not "high enough," a person will not buy a car

$$Y_i = 0 \text{ if } Y_i^* < 0$$

Logit and Probit Models

$$
\begin{aligned}
P_i &= \text{Prob}(Y_i = 1) \\
&= \text{Prob}(Y_i^* \geq 0) \\
&= \text{Prob}(\beta_0 + \beta_1 X_{1i} + \varepsilon_i \geq 0) \\
&= \text{Prob}(\varepsilon_i \geq -\beta_0 - \beta_1 X_{1i}) \\
&= 1 - F(-\beta_0 - \beta_1 X_{1i}) \text{ where } F \text{ is the c.d.f. for } \varepsilon \\
&= F(\beta_0 + \beta_1 X_{1i}) \text{ if } F \text{ is symmetric}
\end{aligned}
$$

- The basic problem is selecting *F*—the cumulative density function for the error term
  - This is where where the two models differ

Logit Model

- For the logit model we specify

$$\text{Prob}(Y_i = 1) = F(\beta_0 + \beta_1 X_{1i}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_{1i})}}$$

- $\text{Prob}(Y_i = 1) \to 0$ as $\beta_0 + \beta_1 X_{1i} \to -\infty$
- $\text{Prob}(Y_i = 1) \to 1$ as $\beta_0 + \beta_1 X_{1i} \to \infty$
  - Thus, probabilities from the logit model will be between 0 and 1

Probit Model

- In the probit model, we assume the error in the utility index model is normally distributed
  - $\varepsilon_i \sim N(0,\sigma^2)$

$$\text{Prob}(Y_i = 1) = F\left(\frac{\beta_0 + \beta_1 X_{1i}}{\sigma}\right)$$

  - Where *F* is the standard normal cumulative density function (c.d.f.)

$$\text{Prob}(Y_i = 1) = F\left(\frac{\beta_0 + \beta_1 X_{1i}}{\sigma}\right) = \int_{-\infty}^{\frac{\beta_0+\beta_1 X_{1i}}{\sigma}} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$$

The Tobit Model

- Can also have latent variable models that don't involve binary dependent variables
- Say $y^* = \boldsymbol{x\beta} + u$, $u|\boldsymbol{x} \sim \text{Normal}(0,\sigma^2)$
- But we only observe $y = \max(0, y^*)$
- The Tobit model uses MLE to estimate both $\boldsymbol{\beta}$ and σ for this model
- Important to realize that $\boldsymbol{\beta}$ estimates the effect of $\boldsymbol{x}$ on y*, the latent variable, not y

Conceptualizing Censored Data

- What do we make of a variable like “Hersheys chocolate bars consumed in the past year”?

- *For all the respondents with 0 bars, we think of those cases as “left censored from below”.*

- Think of a latent variable for “willingness to consume Hershey bars” that underlies “bars consumed in the past year”. Individuals who most detest Hershey Bars would score a negative number of bars consumed if that were possible.

Censored Regression Models & Truncated Regression Models

- More general latent variable models can also be estimated, say
- $y = \boldsymbol{x}\beta + u$, $u|\boldsymbol{x},c \sim \text{Normal}(0, \sigma^2)$, but we only observe $w = \min(y,c)$ if right censored, or w = max(y,c) if left censored
- Truncated regression occurs when rather than being censored, the data is missing beyond a censoring point

Estimation

- Probability

$$\Pr[Y_i = 0 \mid X_i] = \Pr[X_i\beta + \varepsilon_i \le 0 \mid X_i] = \Pr[\varepsilon_i \le -X_i\beta \mid X_i]$$

$$= \Pr\left[\frac{\varepsilon_i}{\sigma} \le -\frac{X_i\beta}{\sigma} \mid X_i\right] = \Phi\left(-\frac{X_i\beta}{\sigma}\right)$$

$$\Pr[Y_i > 0 \mid X_i] = 1 - \Phi\left(-\frac{X_i\beta}{\sigma}\right)$$

Estimation– see how OLS biased

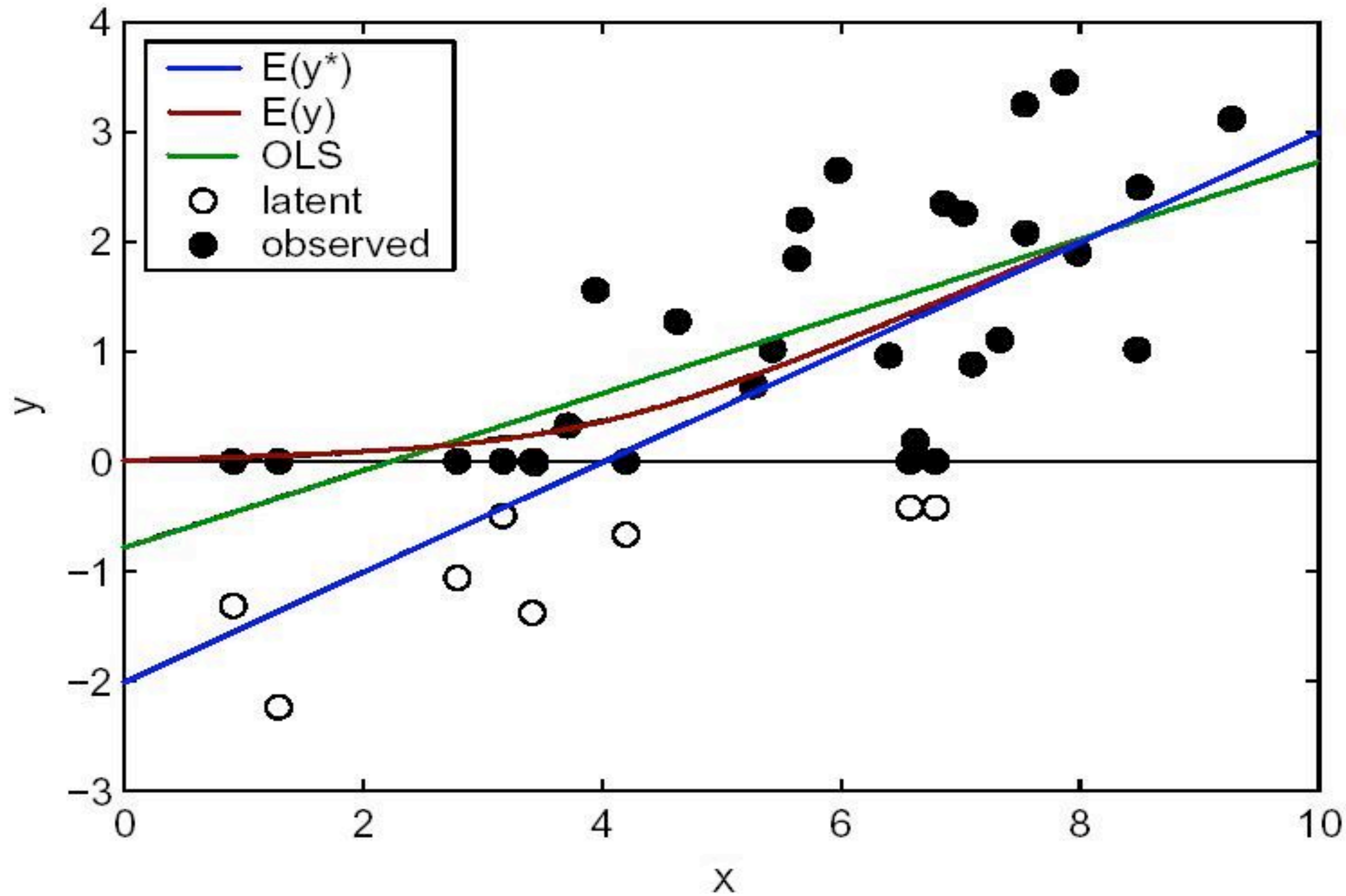
Standard Tobit with:

$$N = 30$$

$$K = 2$$

$$\beta = \begin{bmatrix} -2 \\ 0.5 \end{bmatrix}$$

$$\sigma = 1$$



Empirical Example (Extramarital Affairs)

- *A Theory of Extramarital Affairs*, Ray C.Fair, 1978.
- 601 observations
- Left-censored at 0; Right-censored at 12
- Steps:
  1. Guess the signs of coefficients
  2. Build the final model (compare to our guess)
  3. Who are the most and least “restraint”?

Introduce Variables & Guess signs

- **y_pt** = number of extramarital affairs (annually)

  (0, 1, 2, 3, 7 ~ *4-10 times*, 12 ~ *more than 12)*
- **Z1**= male dummy **(+)**
- **Z2**= age **(-)**
- **Z3**= number of years married **(+)**
- **Z4**= child dummy **(-)**
- **Z5**= How religious **(-)**
- **Z6**= Level of education **(ambiguous)**
- **Z7**= occupation **(ambiguous)**
- **Z8**= marriage satisfaction **(-)**

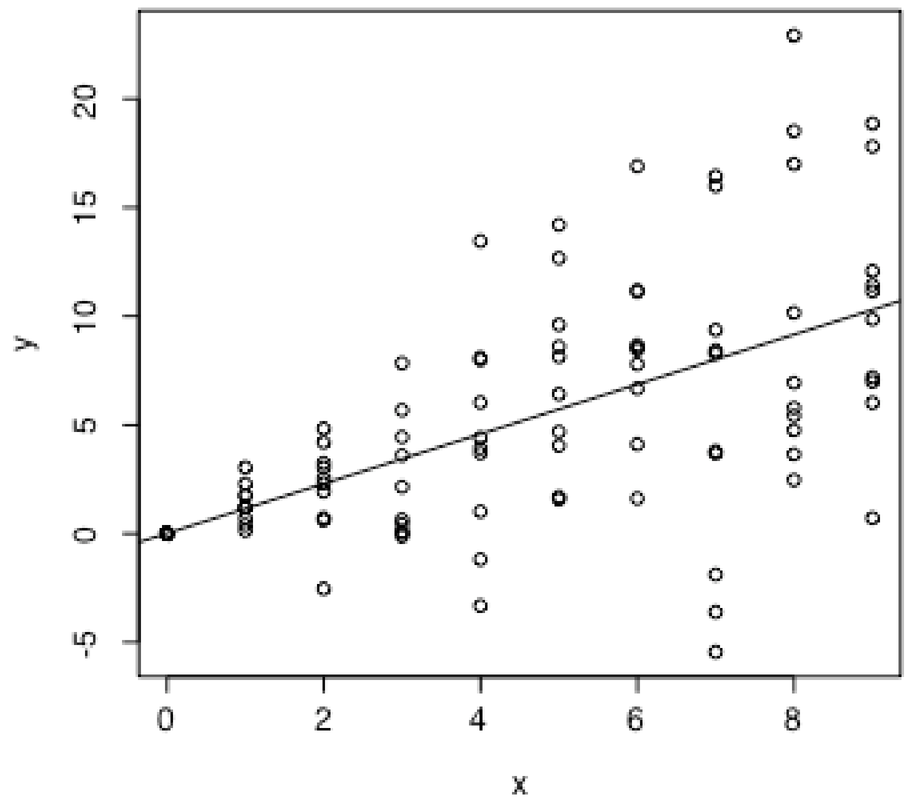
Variables	OLS_{601}	Tobit	OLS_{150}
Constant	5.8860 (0.7753)	9.0829 (2.6588)	8.0963 (1.4254)
Z2 age	-0.0433 (0.0216)	-0.1603 (0.0777)	
Z3 number of years married	0.1551 (0.0367)	0.5389 (0.1342)	0.2179 (0.0651)
Z5 religiousness	-0.4868 (0.1112)	-1.7234 (0.4047)	-0.6999 (0.2844)
Z8 self-rating of marriage	-0.7055 (0.1183)	-2.2673 (0.0408)	-0.6797 (0.2737)

A weakness of the Tobit model

- The Tobit model makes the same assumptions about error distributions as the OLS model, but it is much more vulnerable to violations of those assumptions.
- In an OLS model with *heteroskedastic* errors, the estimated standard errors can be too small.
- In a Tobit model with heteroskedastic errors, the computer uses a bad estimate of the error distribution to determine the chance that a case would be censored, and the coefficient is <u>badly biased.</u>

Data set with heteroskedastic error distribution

- This data set still has Y = X + e, but the range of e increases with X.



OLS with heteroskedastic error

- The OLS regression model still gives good values for the slope and intercept, but you can’t really trust the *t* scores.

```
. * try models with heteroskedastic error terms
. generate e2 = x*zscore
. generate y2 = x + e2
. regress y2 x

  Source |       SS       df       MS              Number of obs =     100
---------+------------------------------           F(  1,    98) =   34.77
   Model |   829.11703     1   829.11703           Prob > F      =  0.0000
Residual |  2337.02002    98  23.8471431           R-squared     =  0.2619
---------+------------------------------           Adj R-squared =  0.2543
   Total |  3166.13705    99  31.9811824           Root MSE      =  4.8834

------------------------------------------------------------------------------
      y2 |      Coef.   Std. Err.       t     P>|t|       [95% Conf. Interval]
---------+--------------------------------------------------------------------
       x |   1.002492   .1700166      5.896   0.000       .6650998    1.339884
   _cons |  -.2737584     .90764     -0.302   0.764       -2.07494    1.527424
------------------------------------------------------------------------------
```

Tobit with heteroskedastic error

- The Tobit model gives values for the slope and intercept that are simply incorrect. (Too steep)

```
Tobit estimates                                   Number of obs   =        100
                                                  LR chi2(1)      =      38.04
                                                  Prob > chi2     =     0.0000
Log likelihood = -175.25383                       Pseudo R2       =     0.0979

------------------------------------------------------------------------------
      y2 |      Coef.   Std. Err.       t     P>|t|       [95% Conf. Interval]
---------+--------------------------------------------------------------------
       x |   1.880274   .3261122      5.766   0.000       1.233196    2.527351
   _cons |  -6.947138     2.0908     -3.323   0.001      -11.09574   -2.798537
---------+--------------------------------------------------------------------
     _se |   6.612864   .7459541            (Ancillary parameter)
------------------------------------------------------------------------------

Obs. summary:         55 left-censored observations at y2<=3
                      45 uncensored observations
```

Summary of Tobit models

- Tobit models can be very useful, but you must be *extremely* careful.
- Graph the relationship between each explanatory variable and the outcome variable. Look for evidence that the errors are not identically and normally distributed.