Sections 6.1, 6.2, 6.3

Timothy Hanson

Department of Statistics, University of South Carolina

Stat 770: Categorical Data Analysis

Should you always toss in a dispersion term ϕ?

Why not always use quasilikelihood (Section 4.7)?

Here's some SAS code:

```
data example;
 input x y n @@; x_sq=x*x;
 datalines;
 -2.0 86 100 -1.5 58 100 -1.0 25 100 -0.5 17 100 0.0 10 100
  0.5 17 100  1.0 25 100
 ;
proc genmod; * fit simple linear term in x & check for overdispersion;
 model y/n = x / link=logit dist=bin;
proc genmod; * adjust for apparent overdispersion;
 model y/n = x / link=logit dist=bin scale=pearson;
proc genmod; * what if instead we try a more flexible mean?;
 model y/n = x x_sq / link=logit dist=binom;
proc logistic; * residual plots from simpler model;
 model y/n = x; output out=diag1 reschi=p h=h xbeta=eta;
data diag2; set diag1; r=p/sqrt(1-h);
proc gplot; plot r*x; plot r*eta;
```

Output from fit of logistic model with logit link

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	5	74.6045	14.9209
Pearson Chi-Square	5	79.5309	15.9062

Analysis Of Parameter Estimates

Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	-1.3365	0.1182	-1.5682	-1.1047	127.77	<.0001
x	1	-1.0258	0.0987	-1.2192	-0.8323	108.03	<.0001
Scale	0	1.0000	0.0000	1.0000	1.0000		

The coefficient for x is highly significant. Note that $P(\chi^2_5 > 74.6) < 0.0001$ and $P(\chi^2_5 > 79.5) < 0.0001$. Evidence of overdispersion? There's good replication here, so certainly *something* is not right with the model.

Let's include a dispersion parameter ϕ

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	5	74.6045	14.9209
Scaled Deviance	5	4.6903	0.9381
Pearson Chi-Square	5	79.5309	15.9062
Scaled Pearson X2	5	5.0000	1.0000

Analysis Of Parameter Estimates

Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	-1.3365	0.4715	-2.2607	-0.4123	8.03	0.0046
x	1	-1.0258	0.3936	-1.7972	-0.2543	6.79	0.0092
Scale	0	3.9883	0.0000	3.9883	3.9883		

We have $\sqrt{\hat{\phi}} = 4.0$ and the standard errors are increased by this factor. The coefficient for x is still significant.
Problem solved!!! Or is it?

More flexible mean

Instead of adding ϕ to a model with a linear term, what happens if we allow the mean to be a bit more flexible?

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	4	1.7098	0.4274
Pearson Chi-Square	4	1.6931	0.4233

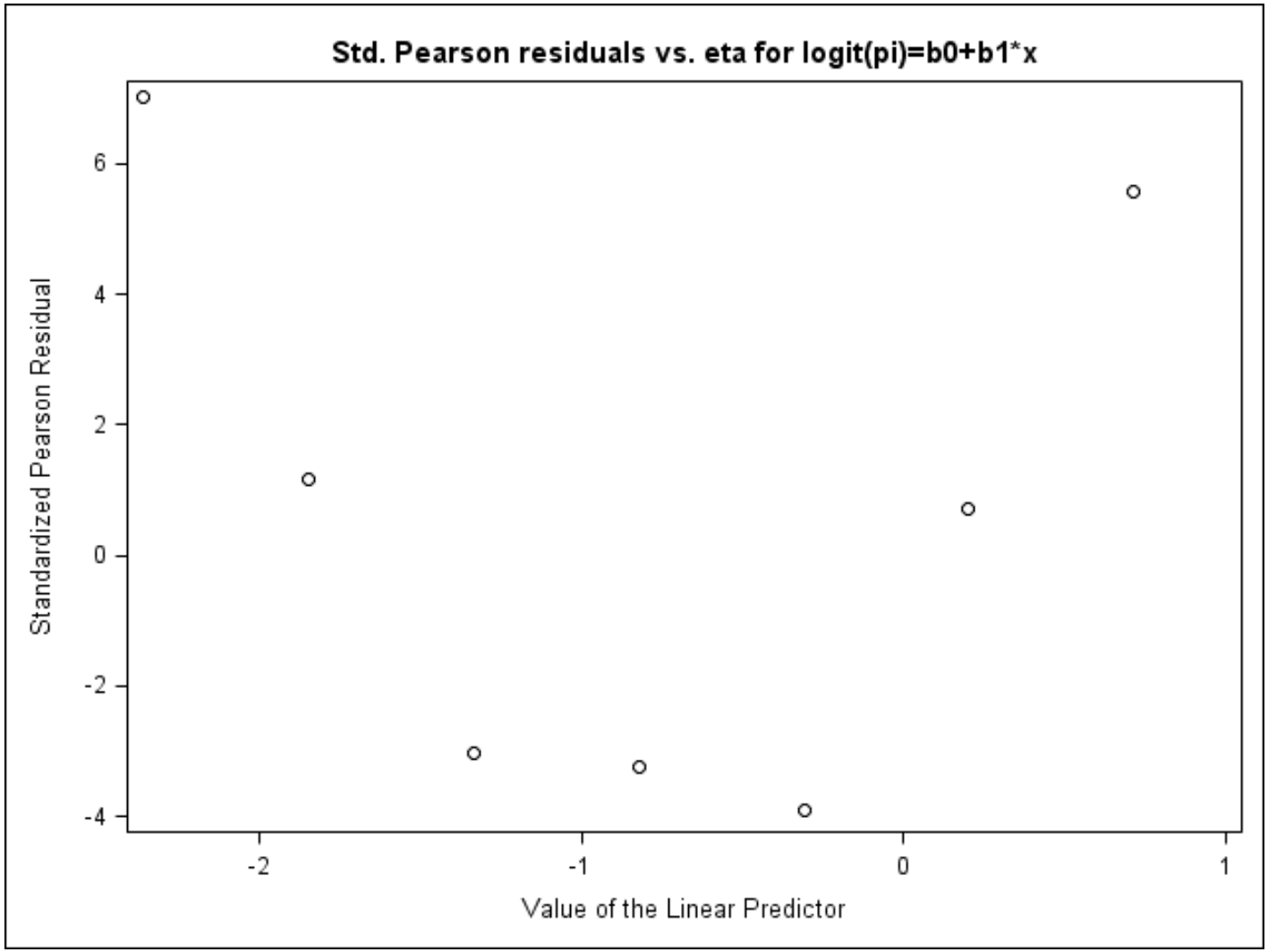
Analysis Of Parameter Estimates

Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	-1.9607	0.1460	-2.2468	-1.6745	180.33	<.0001
x	1	-0.0436	0.1352	-0.3085	0.2214	0.10	0.7473
x_sq	1	0.9409	0.1154	0.7146	1.1671	66.44	<.0001
Scale	0	1.0000	0.0000	1.0000	1.0000		

Here, we are *not* including a dispersion term ϕ. There is no evidence of overdispersion when the *mean is modeled correctly*. Adjusting SE's using the quasilikelihood approach relies on *correctly modeling the mean*, otherwise ϕ becomes a measure of dispersion of data about *an incorrect mean*. That is, ϕ attempts to pick up the slop left over from specifying a mean that is too simple.

Residual plot r_i versus $\hat{\eta}_i$ for made-up data.

A correctly specified mean can obviate overdispersion. How to check if the mean is okay? Hint:



6.1 Model selection

Two competing goals:

- Model should fit the data well.
- Model should be simple to interpret (smooth rather than overfit – principle of parsimony).

Often hypotheses on how the outcome is related to specific predictors will help guide the model building process.

6.1.1: Agresti points out a rule of thumb: at least 10 events and 10 non-events should occur for each predictor in the model (including dummies). So if $\sum_{i=1}^{N} y_i = 42$ and $\sum_{i=1}^{N}(n_i - y_i) = 830$, you should have no more than $42/10 \approx 4$ predictors in the model.

6.1.2 Horseshoe crab data

Recall that in all models fit we strongly rejected $H_0 : \text{logit } \pi(\mathbf{x}) = \beta_0$ in favor of $H_1 : \text{logit } \pi(\mathbf{x}) = \mathbf{x}'\boldsymbol{\beta}$:

Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	40.5565	7	<.0001
Score	36.3068	7	<.0001
Wald	29.4763	7	0.0001

However, it was not until we carved superfluous predictors from the model that we showed significance for the included model effects. This is an indication that several covariates may be highly related, or correlated. If one or more predictors are perfectly predicted as a linear combination of other predictors the model is overspecified and unidentifiable. Here's an example:

$$\text{logit } \pi(\mathbf{x}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3(x_1 - 3x_2).$$

Multicollinearity

The MLE $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2, \beta_3)$ is not unique and the model is said to be unidentifiable. The variable $x_1 - 3x_2$ is totally predicted and redundant given x_1 and x_2.

Although a perfect linear relationship is usually not met in practice, often variables are *highly* correlated and therefore one or more are redundant. We need to get rid of some!

Although not ideal, automated model selection is necessary with large numbers of predictors. With $p - 1 = 10$ predictors, there are $2^{10} = 1024$ possible models; with $p - 1 = 20$ there are $1,048,576$ to consider.

Backwards elimination starts with a large pool of potential predictors and step-by-step eliminates those with (Wald) p-values larger than a cutoff (the default is 0.05 in SAS PROC LOGISTIC).

6.1.4 Backwards elimination for crab data

```
proc logistic data=crabs1 descending;
 class color spine / param=ref;
 model y = color spine width weight color*spine color*width color*weight
  spine*width spine*weight width*weight / selection=backward;
```

When starting from all main effects and two-way interactions, the default p-value cutoff 0.05 yields only the model with width as a predictor

Summary of Backward Elimination

Step	Effect Removed	DF	Number In	Wald Chi-Square	Pr > ChiSq
1	color*spine	6	9	0.0837	1.0000
2	width*color	3	8	0.8594	0.8352
3	width*spine	2	7	1.4906	0.4746
4	weight*spine	2	6	3.7334	0.1546
5	spine	2	5	2.0716	0.3549
6	width*weight	1	4	2.2391	0.1346
7	weight*color	3	3	5.3070	0.1507
8	weight	1	2	1.2263	0.2681
9	color	3	1	6.6246	0.0849

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-12.3508	2.6287	22.0749	<.0001
width	1	0.4972	0.1017	23.8872	<.0001

Change criteria for removing predictor to p-value ≥ 0.15

```
model y = color spine width weight color*spine color*width color*weight
 spine*width spine*weight width*weight / selection=backward slstay=0.15;
```

Yields more complicated model:

Summary of Backward Elimination

Step	Effect Removed	DF	Number In	Wald Chi-Square	Pr > ChiSq
1	color*spine	6	9	0.0837	1.0000
2	width*color	3	8	0.8594	0.8352
3	width*spine	2	7	1.4906	0.4746
4	weight*spine	2	6	3.7334	0.1546
5	spine	2	5	2.0716	0.3549

Analysis of Maximum Likelihood Estimates

Parameter		DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept		1	13.8781	14.2883	0.9434	0.3314
color	1	1	1.3633	5.9645	0.0522	0.8192
color	2	1	-0.6736	2.6036	0.0669	0.7958
color	3	1	-7.4329	3.4968	4.5184	0.0335
width		1	-0.4942	0.5546	0.7941	0.3729
weight		1	-10.1908	6.4828	2.4711	0.1160
weight*color	1	1	0.1633	2.3813	0.0047	0.9453
weight*color	2	1	0.9425	1.1573	0.6632	0.4154
weight*color	3	1	3.9283	1.6151	5.9155	0.0150
width*weight		1	0.3597	0.2404	2.2391	0.1346

Drop width and width*weight?

Let's test if we can simultaneously drop width and width*weight from this model. From the (voluminous) output we find:

Criterion	Intercept Only	Intercept and Covariates
AIC	227.759	196.841
SC	230.912	228.374
-2 Log L	225.759	176.841

Fitting the simpler model with color, weight, and color*weight yields

Model Fit Statistics

Criterion	Intercept Only	Intercept and Covariates
AIC	227.759	197.656
SC	230.912	222.883
-2 Log L	225.759	181.656

There are 2 more parameters in the larger model (for width and width*weight) and we obtain $-2(L_0 - L_1) = 181.7 - 176.8 = 4.9$ and $P(\chi^2_2 > 4.9) = 0.07$. We barely accept that we can drop width and width*weight at the 5% level.

Forward selection

Forward selection starts by fitting each model with one predictor separately and including the model with the smallest p-value under a cutoff (default=0.05 in PROC LOGISTIC). When we instead have SELECTION=FORWARD in the MODEL statement we obtain the model with only width. Changing the cutoff to SLENTRY=0.15 gives the model with width and color.

Starting from main effects and working backwards by hand, we ended up with width and color in the model. We further simplified color to dark and non dark crabs. Using backwards elimination with a cutoff of 0.05 we ended up with just width. A cutoff of 0.15 and another "by hand" step (at the 0.05 level) yielded weight, color, and weight*color.

Your book considers backwards elimination starting with a three-way interaction model including color, spine condition, and width. The end model is color and width.

Stepwise procedure and SAS options

PROC LOGISTIC allows backwards elimination, forwards selection, and something that does both, termed 'stepwise.'

Stepwise selection checks to see whether one or more effects can be removed from the model after adding a term. Stepwise goes back and forth adding and removing terms until no more can be eliminated at the SLSTAY level and no more can be added at the SLENTRY level. In my opinion, this is the best of the three approaches to variable selection.

Hierarchical models have interactions and/or quadratic effects only when the main effects comprising them are also in the model (more on this shortly). SAS automatically chooses the default HIERARCHY=SINGLE to force a hierarchical final model. There are other options, e.g. HIER=MULTIPLE or HIER=NONE.

Stepwise procedure and SAS options

Recall that default values for SLENTRY and SLSTAY are 0.05. You will get models with more predictors when you increase these.

For default SLENTRY and SLSTAY, only width is picked using all three selection procedures for the crab data. For SLENTRY=SLSTAY=0.1, all three procedures give the same model: color and width.

Treating color and spine as *continuous* also yields an additive model with color and width using all three approaches.

6.1.6 AIC & model selection

"No model is correct, but some are more useful than others." – George Box.

It is often of interest to examine several competing models. In light of underlying biology or science, one or more models may have relevant interpretations within the context of why data were collected in the first place.

In the absence of scientific input, a widely-used model selection tool is the Akaike information criterion (AIC),

$$\text{AIC} = -2[L(\hat{\boldsymbol{\beta}}; \mathbf{y}) - p].$$

The $L(\hat{\boldsymbol{\beta}}; \mathbf{y})$ represents model fit. If you add a parameter to a model, $L(\hat{\boldsymbol{\beta}}; \mathbf{y})$ has to increase. If we only used $L(\hat{\boldsymbol{\beta}}; \mathbf{y})$ as a criterion, we'd keep adding predictors until we ran out. The p penalizes for the number of the predictors in the model.

AIC for crab data models

The AIC has very nice properties in large samples in terms of prediction. The smaller the AIC is, the better the model fit (asymptotically).

Model	AIC
W	198.8
$C+Wt+C*Wt$	197.7
$C+W$	197.5
$D+Wt+D*Wt$	194.7
$D+W$	194.0
C^*+W	196.7

*color continuous,

If we pick one model, it's $W+D$, the additive model with width and the dark/nondark category.

LASSO for logistic regression

SAS has a new procedure, PROC HPGENSELECT, which can implement the LASSO, a modern variable selection technique. I was able to get it running on "cloud SAS". It does not, as of yet, have a HIER=SINGLE option akin to PROC GLMSELECT, but probably will in a future version. SAS will perform forward selection with a very large number of variables in a more principled manner than traditional forward selection in PROC HPGENSELECT with the METHOD=LASSO option. It will star the model with the "best" selection criterion that you ask for, below the AIC corrected for small sample sizes. Here we try to find a parsimonious model from all main effects and two-way interactions.

```
proc hpgenselect;
class color spine;
 model y(event="1") = color spine width weight color*spine color*width
 color*weight spine*width spine*weight weight*width / dist=binary link=logit;
 selection method=lasso(choose=aicc) details=all;
run;
```

6.2 Diagnostics

GOF tests are global checks for model adequacy.
The data are $(\mathbf{x}_i, Y_i)$ for $i = 1, \ldots, N$. The i^{th} *fitted value* is an estimate of $\mu_i = E(Y_i)$, namely $\widehat{E(Y_i)} = \hat{\mu}_i = n_i\hat{\pi}_i$ where $\pi_i = \frac{e^{\beta'\mathbf{x}_i}}{1+e^{\beta'\mathbf{x}_i}}$ and $\hat{\pi}_i = \frac{e^{\hat{\beta}'\mathbf{x}_i}}{1+e^{\hat{\beta}'\mathbf{x}_i}}$. The raw residual is what we see Y_i minus what we predict $n_i\hat{\pi}_i$. The Pearson residual divides this by an estimate of $\sqrt{\text{var}(Y_i)}$:

$$e_i = \frac{y_i - n_i\hat{\pi}_i}{\sqrt{n_i\hat{\pi}_i(1-\hat{\pi}_i)}}.$$

The Pearson GOF statistic is

$$X^2 = \sum_{i=1}^{N} e_i^2.$$

Standardized Pearson residual r_i

$$r_i = \frac{y_i - n_i\hat{\pi}_i}{\sqrt{n_i\hat{\pi}_i(1-\hat{\pi}_i)(1-\hat{h}_i)}},$$

where $\hat{h}_i$ is the i^{th} diagonal element of the *hat* matrix $\hat{\mathbf{H}} = \hat{\mathbf{W}}^{1/2}\mathbf{X}(\mathbf{X}'\hat{\mathbf{W}}\mathbf{X})^{-1}\mathbf{X}'\hat{\mathbf{W}}^{1/2}$ where $\mathbf{X}$ is the design matrix

$$\mathbf{X} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1,p-1} \\ 1 & x_{21} & \cdots & x_{2,p-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{N,p-1} \end{bmatrix},$$

and

$$\hat{\mathbf{W}} = \begin{bmatrix} n_1\hat{\pi}_1(1-\hat{\pi}_1) & 0 & \cdots & 0 \\ 0 & n_2\hat{\pi}_2(1-\hat{\pi}_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & n_N\hat{\pi}_N(1-\hat{\pi}_N) \end{bmatrix}.$$

Alternatively, (6.2.1, p. 216) defines a deviance residual.

Comments

- With good replication (e.g. $n_i \geq 10$, plots of residuals r_j versus one of the $p-1$ predictors x_{ij}, for $j = 1, \ldots, N$ should show randomness, just as in regular regression. If a pattern exists, adding nonlinear terms or interactions can improve fit.
- With truly continuous predictors $n_i = 1$ and the residual plots *will* have a distinct pattern. Use the fact that if the model fits, $E(r_i) \approx 0$ and superimpose a loess fit on top of the residuals. The loess line should be *approximately* straight.
- An overall plot is r_j versus the linear predictor $\hat{\eta}_j = \hat{\boldsymbol{\beta}}' \mathbf{x}_j$. This plot will tell you if the model tends to over or underpredict the observed data for ranges of the linear predictor.
- The r_i are approximately $N(0, 1)$ when n_i is not small.
- Individual r_i might flag some individuals (e.g. crabs) ill-fit by the model; a rule-of-thumb is to flag $|r_i| > 3$ as ill-fit.

Fit $W + D$ to the crab data

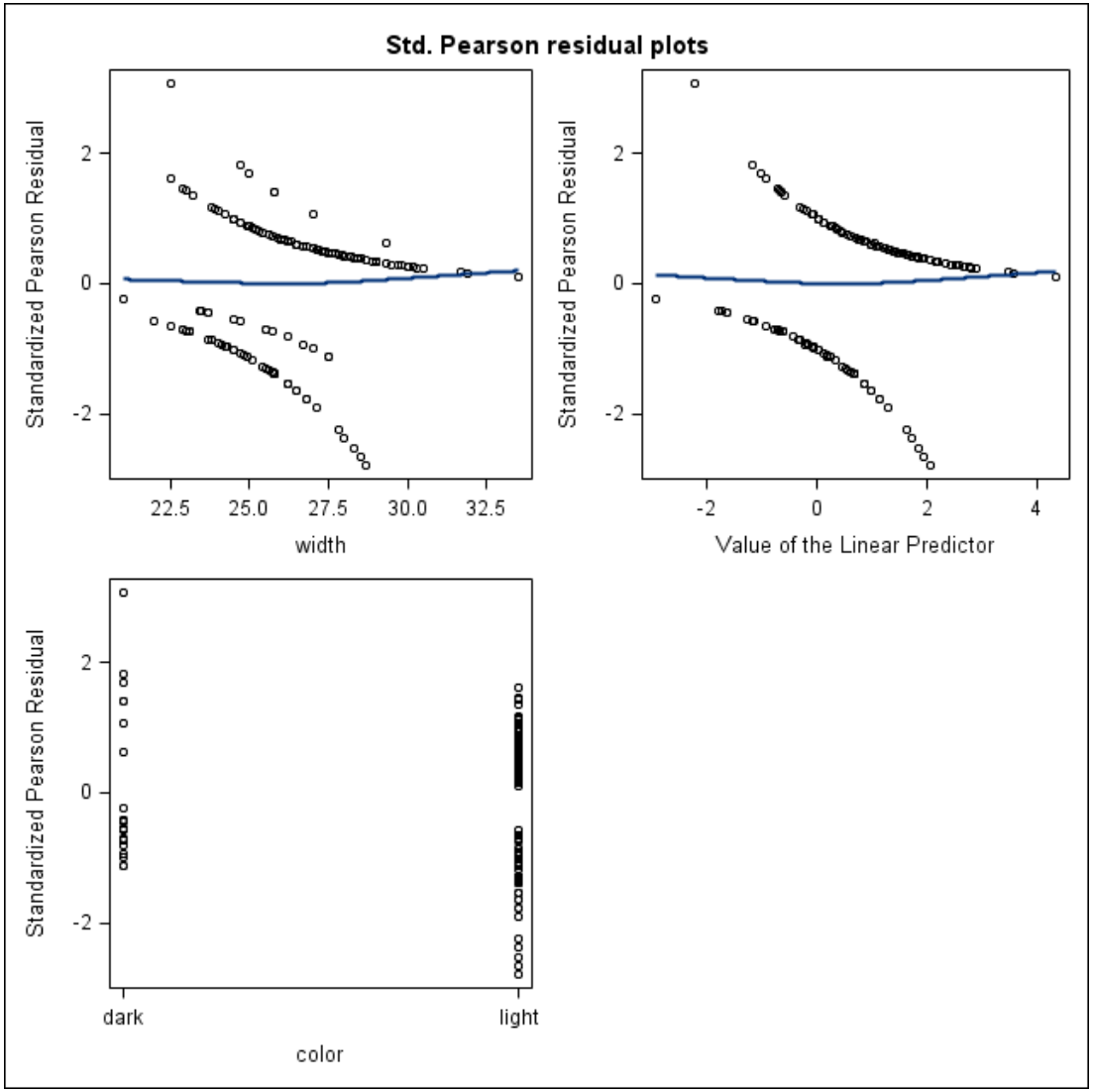
The DATA step is

```
data crabs;
input orig_color spine width satell weight @@; weight=weight/1000; orig_color=orig_color-1;
y=0; if satell>0 then y=1; color='light'; if orig_color=4 then color='dark'; id=_n_;
datalines;
3  3  28.3  8  3050 4  3  22.5  0  1550 2  1  26.0  9  2300 4  3  24.8  0  2100 4  3  26.0  4  2600
3  3  23.8  0  2100 2  1  26.5  0  2350 4  2  24.7  0  1900 3  1  23.7  0  1950 4  3  25.6  0  2150
                                        ...et cetera...
```

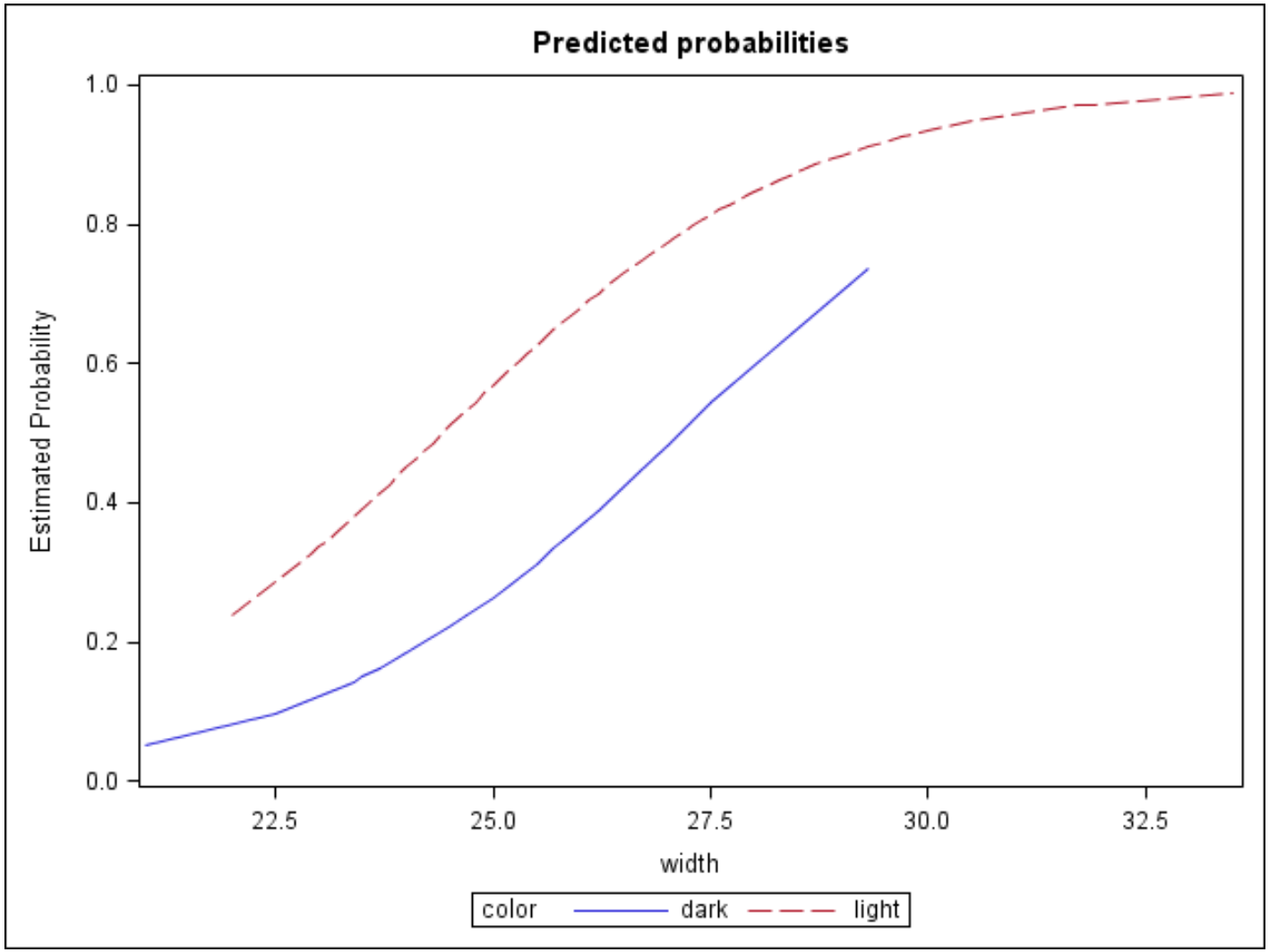
Model fit, r_i plots, and fitted probabilities:

```
proc logistic data=crabs descending; class color / param=ref;
model y=color width;
output out=diag1 stdreschi=r xbeta=eta p=p;

proc sort; by color weight;

proc sgscatter data=diag1;
title "Std. Pearson residual plots";
plot r*(width eta) r*color / loess;
run;

proc sgplot data=diag1;
title1 "Predicted probabilities";
series x=width y=p / group=color;
yaxis min=0 max=1;
run;
```

r_i vs. weight, $\hat{\eta}_i$ and color



Fitted probabilities of one or more satellites



6.2.4 Influence

Unlike linear regression, the leverage $\hat{h}_i$ in logistic regression depends on the model fit $\hat{\boldsymbol{\beta}}$ as well as the covariates $\mathbf{X}$. Points that have extreme predictor values $\mathbf{x}_i$ may not have high leverage $\hat{h}_i$ if $\hat{\pi}_i$ is close to 0 or 1. Here are the influence diagnostics available in PROC LOGISTIC:

- Leverage $\hat{h}_i$. Still may be useful for detecting "extreme" predictor values $\mathbf{x}_i$.
- $c_i = e_i^2 \hat{h}_i / (1 - \hat{h}_i)^2$ measures the change in the joint confidence region for $\boldsymbol{\beta}$ when i is left out.
- DFBETA$_{ij}$ is the standardized change in $\hat{\beta}_j$ when observation i is left out.
- The change in the X^2 GOF statistic when obs. i is left out is DIFCHISQ$_i = e_i^2/(1 - \hat{h}_i)$.

Obtaining influence diagnostics in SAS

I suggest looking at plots of c_i vs. i, and possibly the DFBETA's versus i. One way to get all influence diagnostics is to add INFLUENCE and IPLOTS to the PROC LOGISTIC statement, although this generates *a lot* of output; e.g. data:

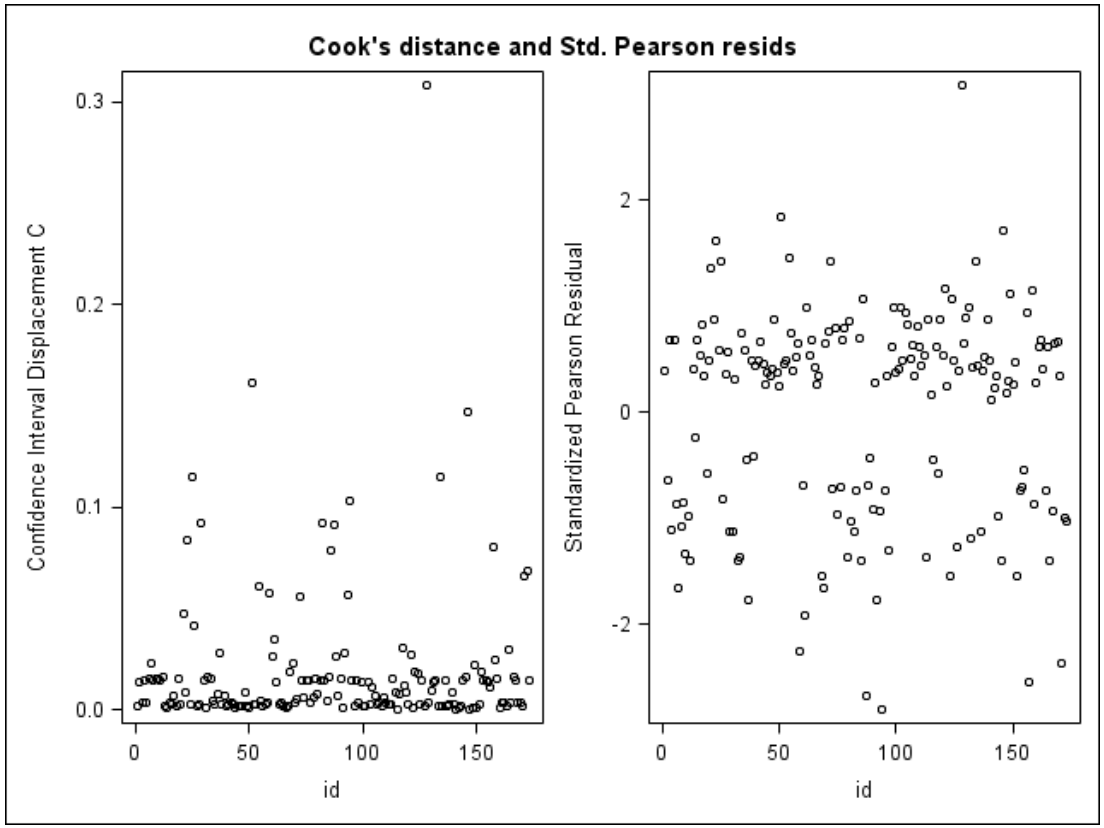
```
proc logistic data=crabs descending; class color / param=ref;
model y=color width / influence iplots;
```

If you add PLOTS to PROC LOGISTIC the output is displayed as plots. Another option is to save the c_i directly and make a plot yourself. Two basic plots to look at are c_i vs. i and r_i vs. i. Note how the ID variable was made in the data step.

```
proc logistic data=crabs descending; class color / param=ref;
model y=color width;
output out=diag2 stdreschi=r c=c;

proc sgscatter data=diag2;
title "Cook's distance and Std. Pearson resids";
plot c*id r*id ;

proc print data=diag2(where=(c>0.2 or r>3 or r<-3));
var y width color c r;
run;
```

Obs	y	width	color	c	r
128	1	22.5	dark	0.30822	3.08247

c_i and r_i vs. i



Obs. 128 has the largest $|r_i|$ and c_i; it is both ill-fit and influential. This is a skinny ($22.5cm$) dark crab that has satellite(s). Recall the probability of having a satellite decreases for dark crabs and for skinny crabs. Comment in light of predicted probability plot.

6.3 Assessment of a model's predictive ability

6.3.3 SAS will "predict" each Bernoulli outcome, say $\tilde{y}_i$ based on a fit of the model *without observation i* with the CTABLE option. You can include the proportion of 'successes' in the population, say it's 30%, using PEVENT=0.3. The default for PEVENT is the proportion of successes in the data set.

An observation will be classified as a success if $\tilde{\pi}_i > k$ where k is a cutoff and $\tilde{\pi}_i$ is the predicted probability of success through the model leaving observation i out; use PPROB=k. If PPROB is omitted, SAS will pick a bunch of them and give the correct number of correctly predicted successes (true positives) and the number of correctly predicted failures (true negatives), as well as the sensitivity and specificity for each.

6.3.4 Sensitivity and specificity for different cutoffs k can be combined into a receiver operator characteristic (ROC) curve; the area under this curve is c. OUTROC=name in the MODEL statement and PLOTS in the PROC LOGISTIC statement gives an ROC curve and estimate of c.

Default predictive ability SAS output

Every pair of observations with different outcomes, i.e. every pair (i_1, i_2) where $y_{i_1} \neq y_{i_2}$, is either concordant, discordant, or tied. Assume $y_{i1} = 1$ and $y_{i_2} = 0$. This pair is concordant if $\hat{\pi}_{i_1} > \hat{\pi}_{i_2}$, discordant if $\hat{\pi}_{i_1} < \hat{\pi}_{i_2}$, and tied if $\hat{\pi}_{i_1} = \hat{\pi}_{i_2}$. Let C be the number of concordant pairs, D the number of discordant pairs, T the number of ties. The total number of pairs is $C + D + T$. Then $\hat{\gamma}$ is $(C - D)/(C + D)$ and Somer's D is $(C - D)/(C + D + T)$. $\hat{\gamma}$ does not penalize for ties. All of this information is in SAS's "Association of Predicted Probabilities and Observed Responses".

c is $(C + 0.5T)/(C + D + T)$: the probability that a randomly drawn "success" will have a higher $\hat{\pi}$ than a randomly drawn "failure", also called "the area underneath the ROC curve." $c \approx 1$ indicates excellent discriminatory ability; $c \approx 0.5$ means you might as well flip a coin rather than use the model to predict success or failure.

The probabilities $\hat{\pi}_i$ are different than the leave-one-out values $\tilde{\pi}_i$ used in the CTABLE option.

$D + W$ in SAS

```
proc logistic data=crabs1 descending plots;
 class dark / param=ref ref=first;
 model y = dark width / outroc=out;
proc logistic data=crabs1 descending plots;
 class dark / param=ref ref=first;
 model y = dark width / ctable;
run;
```

Response Profile

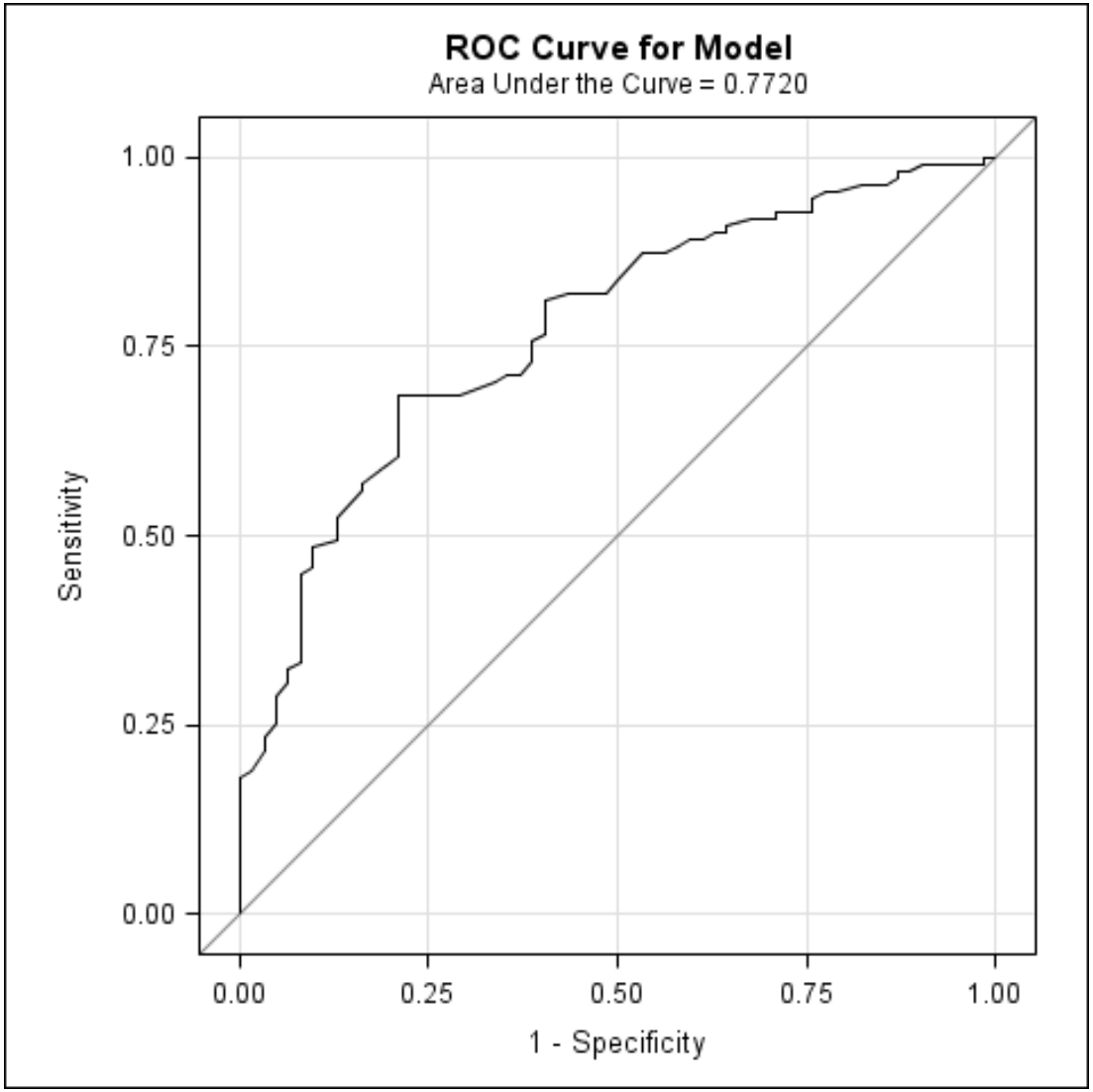
Ordered Value	y	Total Frequency
1	1	111
2	0	62

Association of Predicted Probabilities and Observed Responses

Percent Concordant	76.7	Somers' D	0.544
Percent Discordant	22.3	Gamma	0.549
Percent Tied	0.9	Tau-a	0.252
Pairs	6882	c	0.772

Note that $111 \times 62 = 6882$.

ROC curve for $D + W$



Classification table

	Correct		Incorrect		Percentages				
Prob Level	Event	Non-Event	Event	Non-Event	Correct	Sensi-tivity	Speci-ficity	False POS	False NEG
0.040	111	0	62	0	64.2	100.0	0.0	35.8	.
0.060	111	1	61	0	64.7	100.0	1.6	35.5	0.0
0.080	110	1	61	1	64.2	99.1	1.6	35.7	50.0
0.100	110	1	61	1	64.2	99.1	1.6	35.7	50.0
0.120	110	1	61	1	64.2	99.1	1.6	35.7	50.0
0.140	110	1	61	1	64.2	99.1	1.6	35.7	50.0
0.160	110	3	59	1	65.3	99.1	4.8	34.9	25.0
0.180	110	5	57	1	66.5	99.1	8.1	34.1	16.7
0.200	110	5	57	1	66.5	99.1	8.1	34.1	16.7
0.220	109	5	57	2	65.9	98.2	8.1	34.3	28.6
0.240	108	6	56	3	65.9	97.3	9.7	34.1	33.3
0.260	108	8	54	3	67.1	97.3	12.9	33.3	27.3
0.280	107	8	54	4	66.5	96.4	12.9	33.5	33.3
				...et cetera...					
0.700	63	49	13	48	64.7	56.8	79.0	17.1	49.5
0.720	61	52	10	50	65.3	55.0	83.9	14.1	49.0
0.740	57	54	8	54	64.2	51.4	87.1	12.3	50.0
0.760	54	54	8	57	62.4	48.6	87.1	12.9	51.4
0.780	51	56	6	60	61.8	45.9	90.3	10.5	51.7
0.800	47	57	5	64	60.1	42.3	91.9	9.6	52.9
0.820	39	57	5	72	55.5	35.1	91.9	11.4	55.8
0.840	34	57	5	77	52.6	30.6	91.9	12.8	57.5
0.860	28	59	3	83	50.3	25.2	95.2	9.7	58.5
0.880	21	60	2	90	46.8	18.9	96.8	8.7	60.0
0.900	13	62	0	98	43.4	11.7	100.0	0.0	61.3
0.920	11	62	0	100	42.2	9.9	100.0	0.0	61.7
0.940	5	62	0	106	38.7	4.5	100.0	0.0	63.1
0.960	3	62	0	108	37.6	2.7	100.0	0.0	63.5
0.980	1	62	0	110	36.4	0.9	100.0	0.0	64.0
1.000	0	62	0	111	35.8	0.0	100.0	.	64.2

More detail for ctable

PEVENT is π. For a given k,

	$\tilde{\pi}_i > k$	$\tilde{\pi}_i < k$
$y_i = 1$	n_{11}	n_{10}
$y_i = 0$	n_{01}	n_{00}

Note that n_{++} is the total number of trials.

$$\widehat{se} = \hat{P}(\tilde{\pi}_i > k | y_i = 1) = \frac{n_{11}}{n_{1+}}, \quad \widehat{sp} = \hat{P}(\tilde{\pi}_i < k | y_i = 0) = \frac{n_{00}}{n_{0+}}.$$

$$\begin{aligned} PVP &= P(y_i = 1 | \tilde{\pi}_i > k) \\ &= \frac{P(\tilde{\pi}_i > k | y_i = 1) P(y_i = 1)}{P(\tilde{\pi}_i > k | y_i = 1) P(y_i = 1) + P(\tilde{\pi}_i > k | y_i = 0) P(y_i = 0)} \\ &= \frac{se\ \pi}{se\ \pi + (1 - sp)(1 - \pi)}. \end{aligned}$$

$$\begin{aligned} PVN &= P(y_i = 0 | \tilde{\pi}_i < k) \\ &= \frac{P(\tilde{\pi}_i < k | y_i = 0) P(y_i = 0)}{P(\tilde{\pi}_i < k | y_i = 0) P(y_i = 0) + P(\tilde{\pi}_i < k | y_i = 1) P(y_i = 1)} \\ &= \frac{sp(1 - \pi)}{sp(1 - \pi) + (1 - se)\pi}. \end{aligned}$$